Technical Report

UCAM-CL-TR-839
ISSN 1476-2986

Number 839

UNIVERSITY OF CAMBRIDGE

Computer Laboratory

Automatic extraction of property norm-like data from large text corpora

Colin Kelly

September 2013

15 JJ Thomson Avenue
Cambridge CB3 0FD
United Kingdom
phone +44 1223 763500

http://www.cl.cam.ac.uk/

This technical report is based on a dissertation submitted September 2012 by the author for the degree of Doctor of Philosophy to the University of Cambridge, Trinity Hall.

Technical reports published by the University of Cambridge Computer Laboratory are freely available via the Internet:

http://www.cl.cam.ac.uk/techreports/

ISSN 1476-2986

Summary

Traditional methods for deriving property-based representations of concepts from text have focused on extracting unspecified relationships (e.g., ***car*** – **petrol**) or only a subset of possible relation types, such as hyponymy/hypernymy (e.g., ***car*** *is-a* **vehicle**) or meronymy/metonymy (e.g., ***car*** *has* **wheels**).

We propose a number of varied approaches towards the extremely challenging task of automatic, large-scale acquisition of *unconstrained*, human-like property norms (in the form ***concept*** *relation* **feature**, e.g., ***elephant*** *has* **trunk**, ***scissors*** *used for* **cutting**, ***banana*** *is* **yellow**) from large text corpora. We present four distinct extraction systems for our task. In our first two experiments we manually develop syntactic and lexical rules designed to extract property norm-like information from corpus text. We explore the impact of corpus choice, investigate the efficacy of reweighting our output through WordNet-derived semantic clusters, introduce a novel entropy calculation specific to our task, and test the usefulness of other classical word-association metrics.

In our third experiment we employ semi-supervised learning to generalise from our findings thus far, viewing our task as one of relation classification in which we train a support vector machine on a known set of property norms. Our feature extraction performance is encouraging; however the generated relations are restricted to those found in our training set. Therefore in our fourth and final experiment we use an improved version of our semi-supervised system to initially extract only features for concepts. We then use the concepts and extracted features to anchor an unconstrained relation extraction stage, introducing a novel backing-off technique which assigns relations to concept/feature pairs using probabilistic information.

We also develop and implement an array of evaluations for our task. In addition to the previously employed ESSLLI gold standard, we offer five new evaluation techniques: fMRI activation prediction, EEG activation prediction, a conceptual structure statistics evaluation, a human-generated semantic similarity evaluation and a WordNet semantic similarity comparison. We also comprehensively evaluate our three best systems using human annotators.

Throughout our experiments, our various systems' output is promising but our final system is by far the best-performing. When evaluated against the ESSLLI gold standard it achieves a precision of 44.1%, compared to the 23.9% precision of the current state of the art. Furthermore, our final system's Pearson correlation with human-generated semantic similarity measurements is strong at 0.742, and human judges marked 71.4% of its output as correct/plausible.

Acknowledgements

There are a great many people without whom this thesis would not be in existence. My supervisors, Anna Korhonen and Barry Devereux, have been fantastic teachers and mentors. Anna's patience with me has known no bounds and her wisdom, insight and kindness are inspiring. Barry too has helped me every step of the way: I am extremely grateful to him for his consistent willingness to help, explain and elucidate. I have gained an incredible amount of life and research skills and knowledge from them, and for that I cannot thank them enough.

Thanks must go to my examiners, Stephen Clark and Alessandro Lenci, for engaging so thoroughly with the research. Their comments and feedback have helped strengthen the dissertation whilst giving me pause for reflection. I am also very grateful to Ted Briscoe, Paula Buttery and Ann Copestake for their always-insightful observations and suggestions and I would like to thank the three wise post-docs of NLP, Diarmuid Ó Séaghdha, Andreas Vlachos and Laura Rimell, for their invaluable encouragement and generous dispositions.

Much thanks also to Aurelie Herbelot, James Jardine, Katia Shutova, Joseph Bonneau and everyone else in the south corridor for being friendly faces in the lab over the course of these four years. My gratitude similarly goes to the staff at the Computer Lab for fostering such a positive environment in which to work: special praise must go to Lise Gough who has helped me navigate a multitude of Cambridge sins. Many thanks must also go to all those conference attendees, anonymous reviewers, colleagues and acquaintances in related (and unrelated) fields who provided extremely useful feedback and I am enormously grateful to my 'volunteer' annotators for their time and effort—I cannot thank you enough!

This work would not have been possible without financial support, and for that I am very grateful to the EPSRC, Trinity Hall, the Neil Wiseman Memorial Fund, the Lundgren Fund, and the Cambridge Philosophical Society for funding my research.

Thanks to my fellow MCR members as well as the staff and fellows of Trinity Hall for their part in creating such a wonderful community to work and live in: it truly has been an unforgettable experience. I will always be especially grateful to Emily Floeck, Chris Adriaanse, Chris Thomas, Rachel Linn, Owen Richards, James Thom and Ross

Newton for their support and for making my time at Cambridge memorable, and my sincere thanks go to Emily Mansfield for proof-reading this thesis.

Above all I would like to thank my family for their unwavering support: my sister and brother for keeping my feet on the ground, and my parents for everything they've given and taught me—I dedicate this thesis to them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

THIS PHD IS WRITTEN in the context of an interdisciplinary project that straddles a number of distinct, although inherently linked, fields within the cognitive sciences and computer science: computational linguistics, modelling natural language from a computational perspective; cognitive neuroscience, investigating language function in the human brain; and experimental psycholinguistics, understanding how humans comprehend and process language. The goal of our work is to develop natural language processing (NLP) techniques that will ultimately enable the improvement of property-based models of conceptual structure in experimental psychology.

1.1 Conceptual representation

Humans' mental representation of the world is said to be founded, in part, on concrete concepts such as ***car***, ***zebra*** and ***banana***. The nature of how these representations manifest and express themselves in the brain has been studied extensively in cognitive science, and recent theories of conceptual representation have adopted a distributed, componential and property-based paradigm (e.g., Farah and McClelland, 1991; Randall et al., 2004; Tyler et al., 2000). According to these accounts, concepts are exhibited as patterns of activation across interconnected feature nodes (e.g., *has* **wheels**, *has* **stripes**, *has* **skin**). An important perceived advantage of such models is that they are able to naturally reflect a number of the desirable qualities of a conceptual representation framework. For example, semantic similarity can be intuitively described by way of overlapping patterns of activation. These have been shown to offer predictions consistent with empirical evidence of semantic priming effects (Masson, 1995).

How such concepts manifest and express themselves in the brain has long been viewed as a fundamental question in cognitive neuroscience, and to test these theories, cognitive psychologists (e.g., Randall et al., 2004; Cree et al., 2006; Grondin et al., 2009) have recently moved towards employing empirically grounded, real-world knowledge

to instantiate their models of conceptual representation. To date, such knowledge has principally been derived from property norming studies in which a large number of human volunteers write lists of properties (or 'property norms') of concepts.

For our work, we define a property norm for a concept as a common-sense, descriptive and salient statement characterising that concept. This characterisation can take many forms: for example, an intrinsic quality of the concept (e.g., ***lion*** *is* **animal**), a behaviour performed by or associated with the concept (e.g., ***lion*** — **roars**), or a property which differentiates it from other, similar concepts (e.g., ***lion*** *has* **mane**). One can often unambiguously determine the identity of a concept using only a small number of its property norms: if we know only that ***X*** *is* **animal**, ***X*** — **roars** and ***X*** *has* **mane** then we can surmise that ***X*** is likely to refer to ***lion***.

McRae et al. (2005) collected such a set of norms, which we call the 'McRae norms.' Such norms have been used extensively in experimental psychology research (e.g., in work on concepts, categorisation and semantic memory). The ability to automatically identify and extract such properties from large text corpora could prove extremely beneficial for any researchers employing property norm information in their investigations by removing the limitations imposed by manually produced norms (e.g., their cost to produce, restricted size, fixed nature).

1.2 Property norms

The McRae norms broadly fall into a ***concept*** *relation* **feature** triple pattern, which usually takes the form <***noun***> <*verb*> <**noun/adjective**>. These norms contain a wide variety of types of information such as location (***knife*** *found in* **kitchens**), colour (***cherry*** *is* **red**), parts (***cup*** *has* **handle**), uses (***ladle*** *used for* **stirring**) and so on. A significant minority of the properties are 'behaviour' properties, expressing activities often or typically undertaken by the concepts. These behaviour properties do not take a *relation* verb (instead their relation is labelled *beh*) and thus usually take the form <***noun***> *beh* <**verb**> (e.g., ***aeroplane*** *beh* **crashes**, ***aeroplane*** *beh* **flies**). Some example properties for two concepts are listed in Table 1.1.

Data from property norming studies suffer a number of shortcomings, which have been examined extensively in the literature (see e.g., Murphy, 2002; McRae et al., 2005). One such weakness is that the human participants often under-report certain properties, even when they are facts presumably known by the participants. For example, although *is* **animal** is listed as a property of the majority of animals appearing in the norms, *beh* **breathes** is listed only as a property for ***whale***. Similarly, *has* **heart** is not reported as a property for any animal concept, even though all participants are likely to have known that animals have hearts. A related issue is inconsistency across highly

boat		***porcupine***	
used on water	21	has quills	26
used for transportation	16	an animal	21
has a motor	15	is small	12
made of wood	13	is brown	10
floats	12	has 4 legs	9
used for fishing	9	has legs	9
has sails	7	a mammal	6
used by people	7	is slow	6
used on oceans	7	is dangerous	5
is small	6	lives in forests	5

Table 1.1: Sample properties from the McRae norms for ***boat*** and ***porcupine*** with their citation frequencies.

related concepts: although *has* **legs** is listed as a property of ***leopard***, it is absent for ***tiger***.

1.3 Our task

The objective of our research is to emulate and complement such norming studies by creating a system capable of automatically and comprehensively extracting these types of properties from textual corpora, using techniques from NLP. The ability to do this would be an enormous help to experimental psychologists: one of the key benefits of using computational techniques is that the output is not limited by the labour-intensive enterprise of having humans manually generate sets of properties for each concept. An automatic generation system would allow psychologists to perform large-scale experiments using property norm data for any concepts of their choosing, no longer relying on a pre-determined set of normed concepts. They would be able to extract an extremely large number of properties, perhaps with varying degrees of relevance and specificity. It is this quantification of the relative relevance/salience of the extracted properties which we view as one of the key benefits of such a system. The ultimate output of such an idealised system will therefore be inherently different from the short lists of true properties produced by humans. We will further discuss the theoretical implications of this later in this thesis.

We note that our task is by nature extremely difficult to evaluate: as the domain is broad and unconstrained, and the ideal output unknown/incomplete (indeed, we are aiming to complement already-known output rather than exactly reproduce it), assessing performance with complete accuracy is highly challenging.

1.4 Primary contributions

This work addresses the challenging task of extracting unconstrained human-like, common sense property norms from large text corpora. The primary contributions of our work are threefold. The first is that we offer a number of techniques for the extraction of such norms and insights into various methodologies which can be used for this particular task. The second is that we comprehensively assess our systems' performance; in addition to classic NLP evaluation techniques, we also present a number of novel evaluation methodologies. Finally, we discuss the theoretical implications of such property extraction in detail, situating the output of these computational linguistic techniques within the broader domain of conceptual and semantic knowledge in the cognitive sciences.

1.5 Notes

1.5.1 Terminology

There is ambiguity in using **feature** to describe just the final term of a ***concept*** *relation* **feature** triple pattern, as the entire pattern (or sometimes just the latter two terms) has in previous literature been called a 'feature'. We adopt the convention that a 'property' describes the full property norm while a 'feature' is comprised only of a single term, **feature**, typically the last word of the corresponding property. For example, in ***porcupine*** *is* **dangerous**, the 'feature' is the single term **dangerous**, while the triple's 'property' corresponds to its relation and feature terms, *is* **dangerous**.

There is further ambiguity in that the defining characteristics of data used in machine learning are also typically known as 'features'. We adopt the convention that these machine learning features be called 'machine learning attributes' or just 'attributes'.

1.5.2 Collaboration

The extraction and clustering code base which we used as a starting point for our experiments in Chapter 3 was written by Nicholas Pilkington and Barry Devereux. The evaluation experiment using fMRI data as described in Section 3.3.4 was done in collaboration with Barry Devereux, who also calculated the correlational statistics presented in Section 3.3.6.

All remaining theoretical, experimental and written work was carried out by the author of this thesis alone.

1.5.3 Publications

The majority of the research in this thesis was presented at a number of NLP and psycholinguistic conferences and in a journal article (to appear). The relevant publications are the following:

- *Automatic extraction of property norm-like data from large text corpora* (to appear). Colin Kelly, Barry Devereux, Anna Korhonen. In Cognitive Science journal.
- *Minimally supervised learning for unconstrained conceptual property extraction* (2013). Colin Kelly, Anna Korhonen, Barry Devereux. In Proceedings of the 35th Annual Conference of the Cognitive Science Society.
- *Semi-supervised learning for automatic conceptual property extraction* (2012). Colin Kelly, Barry Devereux, Anna Korhonen. In Proceedings of the 3rd Workshop on Cognitive Modeling and Computational Linguistics.
- *Acquiring human-like feature-based conceptual representations from corpora* (2010). Colin Kelly, Barry Devereux, Anna Korhonen. In Proceedings of the NAACL HLT 2010 First Workshop on Computational Neurolinguistics.
- *Using fMRI activation to conceptual stimuli to evaluate methods for extracting conceptual representations from corpora* (2010). Barry Devereux, Colin Kelly, Anna Korhonen. In Proceedings of the NAACL HLT 2010 First Workshop on Computational Neurolinguistics.

1.6 Thesis outline

The rest of this thesis is divided as follows. In Chapter 2, we offer a survey of the literature relevant to the topic; we first cover the theoretical background supporting our task and offer a more in-depth look at property norms themselves, as well as describing previous work in NLP on tasks which are similar to, and the same as, our own.

In Chapter 3 we describe our pilot experiments, explaining the initial triple extraction technique as well as presenting a number of new evaluation methodologies: we compare our system's output with fMRI activation patterns, introduce a conceptual structure statistics evaluation, directly evaluate against a gold standard, perform a qualitative analysis of the output and execute an EEG evaluation task. In Chapter 4 we describe a blind-trained, comprehensive extraction system, extending our pilot system with a number of key improvements, including a novel reweighting measure. This chapter also introduces a comprehensive human evaluation, as well as two additional

semantic similarity evaluation techniques based on WordNet- and human-generated similarity measures.

In Chapter 5 we introduce an approach which builds on the findings from our previous two experiments. We adjust the system to introduce the semi-supervised learning of rules, viewing the task as one of relation classification and feature selection. In Chapter 6 we describe our final experiment which harnesses and perfects our reasonably performing semi-supervised feature extraction method and employs chunked corpus data to allow for the extraction of unconstrained relations.

Finally, in Chapter 7, we present conclusions and directions for future research.

Chapter 2

Background

IN THIS CHAPTER, we review the literature relevant to our task. We first survey some theoretical aspects of knowledge representation, situating them within the context of property norming studies, knowledge in corpora and data in the brain, and we examine how these various representations might guide us in our application of NLP techniques to this task. We next discuss the motivations for, and some issues associated with, property norming studies. Finally, we offer an in-depth survey of research relevant to our goals within NLP, considering and critiquing how other researchers have approached both similar and identical tasks to our own. We conclude by briefly outlining proposed directions for our experiments.

2.1 Theoretical background

We begin by drawing a theoretical distinction between the various types of semantic data that we believe exist. Humans make use of three principle sources of semantically meaningful data: data already in the brain (conceptual knowledge), data in language (both spoken and written) and data in the world (data derived from non-language-based experiences in, and interactions with, the world). The conceptual data already in the brain (if we ignore the possibility of innate semantic knowledge at birth) will have been derived by way of the other two sources, but may also combine with them to produce further conceptual knowledge (inferences). Property norms could be viewed as a window to this conceptual knowledge, even if it is a window which doesn't encapsulate all of the conceptual knowledge (as demonstrated by the shortcomings described below). However, for concrete nouns at least, property norms could also be viewed as a mere transcription of "data in the world": the norms list real-world properties of real-world concepts. That the norms are, by necessity, written in language introduces the notion of them also being in the domain of "data in language". Property norms

lie therefore at the crossroads of the three data sources. This could be why in previous work in cognitive psychology the three have sometimes been conflated.

There is clearly overlap between the three sources of data, and there are key theoretical questions of how similar they are (modulo the data's representation) as well as the degree of overlap between them. Our work strives to go some way towards answering the question of whether the data in language (for which we use corpora as a proxy) is a sufficient vehicle to capture the full scope of conceptual knowledge; can we, given a sufficiently large body of text, generate all the conceptual knowledge that could be found in the human brain? We will return to these questions in the final discussion, but for now we review previous work in this area.

We begin by examining whether what we are aiming to do is in fact realistic, in terms of the extent to which conceptual knowledge may be extracted from text corpora. Andrews et al. (2009) proposed a theory of semantic representation based on a statistical combination of 'experiential' data ("derived by way of our experience with the physical world") using property norms and 'distributional' data (which "describes the statistical distribution of words across spoken and written language") from the British National Corpus (BNC) (Leech, 1992). They stated that in previous literature the contributions of these two types of data had only been considered independently, never simultaneously or in combination. This hypothesis of two distinct and separate types of data is notable, since in this work we hope to create a system able to extract experiential data directly from distributional data.

To test their theory, Andrews et al. (2009) collected two datasets for each type, experiential and distributional, and constructed a model for both. Their experiential dataset was derived from a property norming study and their distributional model was derived from a sampling of term-document co-occurrence statistics from 7,776 texts in the BNC. They generated three training sets: one properties-only, one text-only and one combined. Their representations were based on the Latent Dirichlet Allocation (LDA) model (Blei et al., 2003) which uses a bag-of-words approach (and therefore disregards sentence syntax and word order). They hoped to discover latent statistical patterns in their training sets and predicted that their experiential data set would produce correlated 'feature-clusters', while their distributional data set would produce clusters of correlated words showing latent 'discourse topics'. Finally, they hoped that models trained on the combined data set would show correlations between the experiential and distributional data sets. Their key hypothesis was that "the latent patterns in each model represent its semantic knowledge", and they wanted to test whether the semantic representation of any word can be understood as (a) a distribution over discourse topics (distributional model), (b) a distribution over feature-clusters (experi-

ential model) or (c) a coupled distribution over both. They evaluated their experiential and distributional models independently and in combination using six datasets offering semantic similarity measurements. They found that their coupled model consistently outperformed both of the constituent models alone.

In a similar vein, Steyvers (2010) augmented probabilistic topic models derived from a text corpus with information from semantic property norms, generating 'feature-topics' which were able to predict missing words in documents, again motivating the combination of a corpus and known property norms to discover semantic information. Andrews and Vigliocco (2010) have also more recently employed word-order information in the framework of hidden Markov Models to further enhance their model. Although the results of Andrews et al. (2009) imply that not all the information we seek will lie in text corpora, we believe that their and Steyvers' research motivates the use of pre-existing property norm data in addition to a large text corpus when searching for further properties. We believe that syntactic information, describing the underlying linguistic structure of sentences, could be a rich resource for identifying property norm-like information from corpora (even if the corpora don't contain *every* desired property). We therefore feel this avenue warrants further investigation.

2.2 Property norming studies

As alluded to above, conceptual representations form the building blocks of one's understanding of the world—they capture the wide variety of information we perceive, ranging from concrete objects to abstract ideas and actions, and how all of these interrelate. There has been widespread interest in cognitive neuroscience concerning the way in which the brain organises these semantic representations and how they function together to produce semantic knowledge. This work has been particularly focused on concrete concepts. As already mentioned, many recent theories in cognitive psychology posit that semantic knowledge is stored in a distributed network of linked property units (Farah and McClelland, 1991; Tyler et al., 2000; Pexman et al., 2002; Randall et al., 2004)—indeed "componentiality is now quite widely assumed in the psycholinguistic literature" (Moss et al., 2007).

To empirically test these distributed, property-based accounts (e.g., Randall et al., 2004; Tyler et al., 2000), in which conceptual representations consist of patterns of activation over sets of interconnected and semantically related property nodes (e.g., *has* **eyes**, *has* **ears**, *is* **large**), cognitive psychologists require an accurate estimate of the kinds of knowledge that people are likely to represent in such a system. The most important sources of such knowledge are property-norming studies. Being able to au-

tomatically extract these types of properties would enable cognitive psychologists to perform large-scale experiments using property-norm data for any concepts of their own choosing, appropriate to their desired task. It would also mitigate the likelihood that not all properties were cited (correctly) for those concepts.

A property norm database is constructed by asking a number of human participants to list properties which they deem important for each given concept. From this it is possible to compile, for each concept, a list of properties with their production frequencies.[1] Such norming studies have been used for implementing and testing models of conceptual representation, experimenting with various accounts of distributed conceptual knowledge in psycholinguistic studies (McRae et al., 1997; Randall et al., 2004; Cree et al., 2006; Tyler et al., 2000; Grondin et al., 2009).

A number of such property norm databases have been constructed (Rosch and Mervis, 1975; Devlin et al., 1998; Ashcraft, 1978; Vinson and Vigliocco, 2008; Garrard et al., 2001; Moss et al., 2002), but few are publicly available. McRae et al. (2005) produced a freely available collection, the largest of its kind, which has been widely used in cognitive modelling and experiments. It offers a comprehensive set of norms for us to train and evaluate our systems on.

In a typical property norming study, subjects will be asked to write down statements which describe a concept (for example, ***aeroplane***). Participants will normally offer phrases such as:

1. *Aeroplanes are found in airports.*
2. *Aeroplanes carry passengers.*
3. *Pilots fly aeroplanes.*
4. *Aeroplanes can crash.*

These statements could be paraphrased into ***concept*** *relation* **feature** triples, each denoting a property of the concept ***aeroplane***. For example, the following properties appear in the McRae norms:

1. ***aeroplane*** *found in* **airports**
2. ***aeroplane*** *used for* **passengers**
3. ***aeroplane*** *requires* **pilots**
4. ***aeroplane*** – **crashes**

[1] i.e., the number of times they were cited as a property for that concept.

The McRae norms were acquired as follows: around 725 participants were asked to produce properties for 541 living and non-living concepts (ranging from ***accordion*** to ***zebra***) with relations linking the concepts to the features. Each participant was asked to list properties of the concept in question, and offered ten blank lines to fill in. They were requested to offer different types of properties (e.g., perceptual properties,[2] functional properties,[3] and other facts[4]). Subjects were each asked to give properties for between 20 and 24 concepts, and each concept had exactly 30 participants listing properties for it. The collectors tried to ensure that subjects offered properties for at most two concepts which the collectors deemed semantically similar—this was to avoid explicit comparisons between similar concepts. Only properties which were listed by at least five participants were included in the final published set.

A notable aspect of the final concatenation of the property norms across the participants was that the collectors decided to ensure that synonymous properties were recorded in the same way, both within and among concepts. They justified this by pointing out that synonymous properties for a given concept are not always enunciated by humans in identical ways (for example, *used for transport*, *is used for transport*, *used for transportation*, *people use it for transport* could each be cited by participants for the same concept). Therefore such properties were normalised to a single representation, with the McRae interpretation of what was considered 'synonymous' made conservatively: "all but the most obvious interpretations [were] verified by multiple colleagues" (McRae, 2012). However, not all details of the process were given. For example, we are unaware of how much variation there was overall, nor do we know exactly how this variation was dealt with. McRae et al.'s final set after this normalisation included a total of 6,996 properties (2,342 unique), with a mean of 13.7 properties per concept.

Although such norms are widely employed in the cognitive sciences, their usefulness has been limited by the fact that these collections are expensive to develop and never achieve anything approaching comprehensiveness. In theory, the advantages of using NLP techniques to extract properties automatically from text corpora are numerous. Corpus-based models offer a better insight into how language is used in practice, as well as providing access to large-scale frequency information. Automatic techniques could discover novel properties (for example, valid properties currently absent in a given property norm set) and could also be used to create norms for other language domains. Since such techniques are less expensive than using humans they could easily be applied to different corpora and used to examine, for example, the effect of domain

[2] e.g., how it looks, feels, sounds, smells, tastes.
[3] e.g., what it does, what it is used for, where and when it is used.
[4] e.g., encyclopedic knowledge: where it is from, which category it would belong to.

variation on the generated properties.

However, property extraction is an extremely challenging NLP task—there is massive variation across the features and relations we seek, and this variation manifests itself in a variety of ways. For one, these properties are completely unconstrained: there are no limitations on what constitutes a semantic property of the kind that might appear in the McRae norms. The nature of properties for a given concept is often dependent on the nature of the concept itself: for example, animals and foods take different 'types' of properties—although both might be described in terms of their appearance (e.g., ***tiger*** *has* **stripes**, ***aubergine*** *is* **purple**), animals would also generally have body parts (e.g., ***cheetah*** *has* **legs**) as well as things they prototypically 'do' (e.g., ***cats*** **–purr**) listed, while foods would have taste and other gustatory sensations as properties (e.g., ***cherry*** *is* **delicious**, ***cucumber*** *is* **crunchy**). Tools, on the other hand, will usually have properties describing their make-up and functions (***umbrella*** has *used for* **keeping dry** and *made of* **plastic** listed as properties). Furthermore, distinctive properties are periodically cited for concepts, and their distinctive nature makes it wholly possible that the listed relation and feature would not apply to any other concept—therefore making them very difficult to learn how to extract. For example, *found at the end of* **a gun** is a listed property for ***bayonet*** in the McRae norms, yet is highly unlikely to apply to any other noun concept. This example also illustrates the issue of property complexity: some properties encapsulate simple relations (e.g., ***apples*** *are* **green**) while others are far more complicated. Ignoring this additional information would mean losing potentially distinctive/key features. For example, *has* **neck** is a shared feature of most animals, but *has* **long neck** is a distinctive property of very few animals (e.g., ***giraffe***).

Even lexically identical relations can have different semantic implications. For example, the meaning of *found in* in ***grasshoppers*** *found in* **summer** differs from its meaning in ***kettles*** *found in* **kitchens**. And as mentioned earlier, there are often many ways to express the same information: ***oven*** *can be* **hot**, ***oven*** *is* **hot** and ***oven*** *gets* **hot** are classified as synonymous properties within the norms. Furthermore, relations can be idiosyncratic: one example is ***bomb*** *dropped from* **aeroplanes**—it is the only norm in the McRae set which has the phrase *dropped from* as its relation. This makes fair evaluation difficult, because of the potential to extract perfectly valid properties which may not appear in our 'correct' set of gold standard norms. Unconstrained property discovery is notoriously difficult and these issues make our task much more challenging compared to related NLP tasks (e.g., ontology extraction).

2.3 Related work

Although a number of approaches to our task currently exist, and although these methods have achieved promising results, these have usually been achieved by limiting the scope of acquisition and/or evaluation.

We first discuss related work in the domains of semantic similarity/relatedness, ontology learning and common sense knowledge extraction. Next, we discuss restricted approaches to our problem, which work only for concept classification rather than concept description. Finally, we discuss more recent research aiming to discover the precise and explicit nature of the relationships between words (and the concepts which they represent). We conclude with two such approaches which together constitute the state of the art. By reviewing this previous work, we hope to use some of these prior techniques as inspiration towards creating our own system.

2.3.1 Semantic similarity and semantic relatedness

Early work on semantics focused on finding semantically similar or related words: one of the first hypotheses in NLP was that word meaning could be modelled as a high-dimensional vector, where the vectors are derived from a corpus. Budanitsky and Hirst (2006) provide an overview of the literature in this area, but techniques employed include using classic co-occurrence vector space models such as Latent Semantic Analysis (Deerwester et al., 1990), Hyperspace Analogue to Language (Lund and Burgess, 1996) and Latent Dirichlet Allocation (Blei et al., 2003) as well as WordNet similarity measures (e.g., Sussna, 1994; Leacock and Chodorow, 1998; Resnik, 1995; Lin, 1998; Jiang and Conrath, 1997). For example, Deerwester et al.'s seminal LSA works by deriving high-dimensionality vectors from collections of discrete, segmented documents, creating a normalised co-occurrence matrix in which rows correspond to words and columns to documents, and reducing the dimensionality of these matrices using singular value decomposition. This enables the computation of the similarity of two words using the cosine angle between their reduced-dimensionality vectors.

2.3.2 Ontology learning

Ontology learning aims to semi-automatically acquire concepts and relations from a corpus with a view to placing them within a given ontology. It is usually limited to a specific domain of interest and requires a formalised representation of the extracted concepts/relations (see, e.g., Buitelaar et al., 2005; Maedche and Staab, 2001; Omelayenko, 2001). A key difference between our work and that of ontology learning is

that while we seek common sense properties of a prototypical nature, ontologists are in search of scientific truths. For example, the McRae norms state that ***tomato*** *is* **a vegetable**, whilst from a botanical perspective, it is a fruit. Similarly in our task we do not, in the first instance, seek properties such as ***tomato*** *contains* **zeaxanthin**. Furthermore, we are not aiming to store our output in a restricted and formalised framework; we are instead hoping to produce a flexible representation of concepts more suited to the conceptual representations from which we derive this task. As already mentioned, properties can take almost any form.

We are also not aiming to extract common sense for the sake of formalising and restructuring it into an 'ontology of everything', as in common sense knowledge extraction (Singh et al., 2002; Lenat, 1995). Rather, we are aiming to extract those most interesting and salient features and relations from a conceptual perspective, whether they be simple or more complicated in structure. We seek that subset of common sense properties which is most important and/or essential to people's knowledge and understanding of a given concept. Such relations and features have traditionally been exemplified by property norms.

2.3.3 Concept/relation classification

There has been a wealth of research in the areas of concept and relation classification. We briefly examine those works which we deem to be most relevant.

Relation classification

Mintz et al. (2009) employed Freebase, a database containing several thousand semantic relations, to train a relation classifier using a paradigm which they called 'distant supervision'. This paradigm assumes that "if two entities participate in a relation, any sentence that contains those two entities might express that relation." For each sentence meeting this criterion, they extracted conjunctive 'textual features' consisting of a number of attributes. These textual features fall into two groupings of attributes: lexical attributes (the sequence of words joining the two entities, their part of speech tags, and a variable window on either side) and syntactic attributes (a dependency path between the entities, and window nodes not in the dependency path). They trained a relation classifier by extracting a large number of textual features for each relation from their corpus. Applying this trained set to their corpus they were able to extract relations between new entities.

This research demonstrates not only the usefulness of employing both lexical and syntactic information but also the potential utility of making use of both labelled and

unlabelled data in a task similar to our own. Their work is distinct from ours in that their algorithm was applied to extracting the most common relations in Freebase, typically well-defined relations between named entities. Their technique also benefited from a relatively large training set of 1.8 million training instances—the McRae norms contain fewer than 7,000 properties.

Shallow concept classification

Barbu (2008) used shallow methods to categorise the McRae norms into semantic classes. He employed a co-occurrence based approach which measured the strength of association between a noun concept and all adjectives and verbs which co-occur with it. He began by classifying the McRae concept properties into six classes on a morphological and semantic level. He then split learning for the property classes into two distinct paradigms. One used a pattern-based approach (four classes) with a seeded pattern-learning algorithm. The other measured strength of association between the concept and referring adjectives and verbs (two classes). His pattern-based approach worked well for properties in the 'superordinate' class, had reasonable recall for the 'stuff' and 'location' classes, but zero recall for 'part' properties. His approach for the other two classes ('quality' and 'action') used four separate association measures (frequency, chi-squared, log-likelihood and pointwise mutual information) which he summed to establish a final score for potential properties. This method yielded good recall and high property precision. He recommended using automated methods for these latter two classes as well as for the superordinate class, but believed a supervised approach was necessary for the other classes due to their more difficult nature.

This method offers the insight that not all features are created equal, and class-dependent methods may be key in extracting features. Indeed the pattern-based approach appears to work well for certain classes. It also motivates a semi-supervised approach for acquiring such features. However as the method is only able to classify the McRae norms, it is not useful for our task of generating new properties. In other words, although his system assesses the performance of its pattern-learning algorithm, it does not postulate the exact nature of the relations being extracted.

Vector-based concept descriptions

Almuhareb and Poesio (2004; 2005) used syntactic patterns to create descriptions of nouns in the form of vector entries. They then built on this by employing not just syntactic patterns but also grammatical relations to create their vector descriptions. They evaluated their approach based on how well their vector descriptions clustered con-

cepts correctly, employing three datasets: the set made by Lund and Burgess (1996); a manually constructed set from WordNet; and their own dataset of 402 nouns, balanced in terms of ambiguity, frequency and class type. They found that their approach performed particularly well at categorising the dataset nouns.

From our perspective, the main issue with their method is that the output again does not yield a property-based description of a particular concept: the vectors they produce contain thousands of features but posit no semantic relationship between those features and the concepts they describe. Comparing these vectors may give a good indication of the extent to which two concepts are similar (in terms of how well they cluster together), but inspecting them doesn't offer the explicitly-stated properties which we seek. However we feel their results do motivate the use of parsed text to improve concept descriptions.

2.3.4 Relation extraction

In NLP, there is already a significant body of work on the topic of relation extraction. As property norming studies aim to gather relationships between entities from humans, relation extraction aims to automatically extract relationships between entities from text corpora.

Lexico-syntactic patterns

Hearst (1992) proposed a lexico-syntactic pattern-based approach for the automatic extraction of hyponyms, and many others have built on her ideas in a variety of ways. Indeed, relation extraction has been used for ontology learning (see Section 2.3.2): Rindflesch et al. (2000) employed NLP methods to extract relationships between cancer therapy genes and drugs from a database of biomedical abstracts, and Pantel and Pennacchiotti (2008) were able to extract specific semantic relations (e.g., *is-a*, *part-of*) from text and link them onto existing semantic ontologies.

Such lexico-syntactic patterns have also been used for tasks such as named-entity classification, where input strings are classified into 'Person', 'Organization' or 'Location' classes. Collins and Singer (1999) used unsupervised learning of relations for this task. In named-entity extraction or ontology learning the relationships and entities are usually well defined: the classes of the words/relations that are sought are closed and there is consequently less ambiguity about whether a particular entity/relation is valid. This relative consistency makes the appearances of entities and relations in corpora more predictable, and renders detection and extraction a far easier task. This in turn means that relatively good performance is achievable through shallow meth-

ods, i.e., methods not requiring deeper syntactic/semantic information such as part of speech or grammatical dependency data (e.g., Etzioni et al., 2005; Poon and Domingos, 2010).

Unsupervised discovery of relationships

Davidov et al. (2007) demonstrated an unsupervised method for discovering instantiations of relationships in which given concepts participate. They used seed data in the form of two or more concept words in a given class (e.g., countries). For each of these concept words, they collected instances of lexical pattern-selected contexts in which the word appeared together with another 'content word' (a word with relatively low frequency in the web-corpus, but high pointwise mutual information with the concept word in question, e.g., capital cities). Similar context groups (e.g., "X is the capital of Y"; "Y's capital is X") were identified across different concept words and merged into single clusters (independent of their respective original concept words). These clusters were used to output sets of concept-target pairs (e.g., Paris–France, Luanda–Angola) representing instantiations of that cluster's relation. This work does not attempt to explicitly name or define the nature of the relationships in question, rather, it merely asserts their existence. We believe that their work motivates an emphasis on patterns which share concept-target instantiations as well as the use of word association measures in the discovery of potential features for our concepts.

Web scale relation extraction

Finally, there has recently been work on the automatic extraction of unbounded binary relations that scale to a web corpus, for example the ReVerb (Etzioni et al., 2011) and WOE (Wu and Weld, 2010) systems. The ReVerb system employs two constraints to identify potential relations with high precision. The first constraint is syntactic; only relation phrases matching a part of speech tag pattern, while the second is lexical; relations with a large number of argument pairs are excluded. These constraints are applied to a very large web-scale corpus to yield a set of relation phrases, and a logistic regression classifier is used to assign confidence scores to each phrase.

Wu and Weld's WOE system constructs training data using heuristic matches between Wikipedia infobox information and corresponding sentences to generate relation-specific examples. It then "abstracts out" these examples to derive relation-independent training data. Wu and Weld implemented their system using both shallow and deep parsing. Of the two, their deep parsing system offered much better performance, which they believed was because the additional syntactic information allowed it to

better handle complicated and/or long-distance relations within sentences.

These systems are designed to extract legitimate relations from a given sentence, with priority given to those relations which the extractor is most confident in. This is similar to but still distinct from our task. Our aim is to capture more general relationships which are 'common sense'; just because an extracted relation is correct in a given context does not automatically make it true in general. Overly specific relations are also less likely to inform the conceptual representation of a concept. That said, we believe Wu and Weld's superior deep parsing results on this general relation extraction task motivates the use of non-shallow techniques in our own experiments.

2.3.5 Extraction of property norm-like descriptions

In summary, we believe our task is more complex than classic relation extraction for three main reasons:

1. We are attempting to simultaneously extract two pieces of information: features of the concept and those features' defining relationship with the concept.

2. The relations which we aim to extract are not limited to a small set of just a few well-defined relations (e.g., *is-a* and *part-of*) nor to the relations of a specific semantic class (e.g., *capital-is* for countries). Indeed the relations can be as many and diverse as the concepts themselves (e.g., each concept could possess a unique and distinguishing relation and feature).

3. We wish to extract those relations and features which would be classified as 'common sense'—those widely understood properties which are easy for humans to recognise but difficult, if not impossible, to describe formally and comprehensively without recourse to human judgements. Furthermore, we seek to prioritise those relations/features which are most salient for describing a concept: these properties are often highly concept-dependent, and can be both distinctive (e.g., ***elephant*** *has* **trunk**) and general (e.g., ***elephant*** *is* **animal**).

In recent years, researchers have begun to develop methods which can automatically extract property norm-like representations from corpora. We feel the development of methods for the extraction of these relations necessitates a much deeper analysis of texts. It also requires a solid grasp of the linguistic phenomena that characterise the conceptually motivated properties we seek. Only Baroni and Lenci (2008), Baroni et al. (2009) and Devereux et al. (2009) have attempted to broach the much more ambitious task of attempting to automatically generate property norm-like data. All have taken their lead from Hearst and her successors, employing manually created rulesets

to extract such properties from corpora. Baroni et al. extracted relational information in the form of 'type-sketches', which gave an approximate, implicit description of the relationship described through "a pattern-based characterization of the relation occurring between a concept and a [feature]", while Devereux et al. aimed to extract explicit relations between the target concepts and their features. The following sections describe both of these approaches in more detail.

Strudel

Baroni and Lenci (2008) introduced an alternative approach to word space models, whereby they searched for semantically meaningful patterns, rather than merely flat co-occurrence of words. They posit that their method can be seen as a "generalization of the pattern-based approach to information mining used by Hearst (1998) and many others." The algorithm, which they call Strudel, works in two stages.

First they take a list of target concepts, and a part of speech tagged corpus, and search for those nouns, verbs and adjectives which are linked to the target concepts by a finite set of 'connector patterns', or templates. How these templates are defined (e.g., target and property are: adjacent, linked by a possessive, connected by a preposition, etc.) dictates what type of relationships are recorded. The second step of their system ranks the concept-property pairs based on the number of distinct linking patterns (rather than the frequency of pattern instances). Their rationale for this states that if a relation is predicated in a number of different ways it is more likely to be an 'interesting' relation, as opposed to a high-frequency relation which is accidental (e.g., idiomatic relations). For example, neither of the phrases "the year of the tiger" and "the tail of the tiger" are uncommon, yet we would not want to extract the triple ***tiger*** *has* **year** as a property of ***tiger***. Therefore we would also want to consider the fact that phrases of the form "the tiger's X tail" are quite common (where X is an adjective), whereas phrases such as "the tiger's X year" are decidedly rare. Their insight here is that "pattern type frequency is a better cue to semantics than token frequency."

The second step of their procedure involves grouping relations linking concepts and properties with similar patterns and creating shallow descriptions of them, noting the distribution of the generalised patterns which they call 'type sketches'. They employ these sketches to remove concept-property pairs whose dominant type is not in the ten most common types in their entire output list (effectively reducing the possibility of unusual relations).

Baroni et al. evaluated Strudel against three other methods—an implementation of Almuhareb and Poesio's attribute-value method (see above), a baseline method based on singular value decomposition and a dependency vector-based method (see Baroni

et al. (2009) for details). They applied the four methods to the 2 billion word UKWAC corpus and evaluated using the ESSLLI dataset which includes 44 concepts from the McRae norms—we describe this test-set in detail in Section 3.3.2. In doing so, they were the first to evaluate directly against the McRae norms. However, as already mentioned, they only evaluated against the features—their produced 'type sketches' are not directly comparable with the relations found in the norms. Of the models they tested, the Strudel method gave highest precision at 23.9%.

Devereux et al.

Devereux et al. (2009) presented a two-stage large-scale property extraction system, which employed class-based semantic information to guide its extraction. It was the first method to focus not only on finding features of concepts, but also on predicting the relation labels between those concepts and features. In their work, Devereux et al. specifically aimed to investigate the usefulness of three types of external information for the task: 1) encyclopedic, 2) syntactic and 3) semantic. The main hypotheses of their method were:

1. It is possible to extract potential features for a given concept from a text corpus using syntactic information, and an intelligent choice of feature-extraction method (e.g., grammatical relation rules) can greatly reduce the amount of noise generated.

2. Using an encylopedic corpus will increase the likelihood of retrieving relevant, property norm-like features.

3. Semantically motivated techniques should be employed to ascertain which features are most relevant to a particular concept.

The first stage (the candidate feature extraction step) of this method was relatively simplistic: the RASP parser (Briscoe, 2006) was used to generate syntactic information (grammatical relations and parts of speech) for each sentence in the corpus. Using this list, one could follow paths (beginning at the concept in question) indicated by the list to generate candidate features of that concept: any and all nouns and adjectives path-linked to the concept by way of a non-auxiliary verb were considered to be potential features. The linking verb was returned as the corresponding 'relation' for the candidate concept-relation-feature triple.

Their relation/feature extraction did not take into account lexical or syntactic constructions that would typically be suggestive of features of the type we are aiming to extract.

Their second stage (the feature reranking step) employed class-based semantic information to upweight relevant features: they performed a semantic analysis of pre-existing feature production norms to reveal information about co-occurring concept and feature classes. Then, using conditional probability calculations, they produced for each concept a ranked list of relation-feature pairs.

As an exploratory methods paper, their system was particularly focused on generating high recall. This was reflected both in their evaluation methodology (considering the top 25% of their returned pairs, even for very large sets of pairs) and their best reported F-scores: 0.126 when matching on features only, and 0.044 when matching on features and relations against the McRae norms (with recall scores of 0.317 and 0.116 respectively). However, they also found that for one of their systems, 38% of their 'incorrect' pairs were judged as correct or plausible when evaluated by human judges. This indicates that such direct evaluation against a gold standard is, at best, incomplete.

We believe there are many components of Devereux et al.'s system which could be improved upon. For example, they did not employ statistical association measures which could similarly improve performance, such as those used in the Baroni and Lenci (2008) method. Furthermore, the two stages were conducted independently of one another—syntactic information acquired from the parsing was lost in the subsequent reranking stage. And because of the way the initial extraction method was implemented, it effectively amounted to a co-occurrence approach.

We believe their underlying hypotheses make sense and the flexibility of their system provides a good starting point for our task.

2.4 Summary

These two final models constitute the state of the art in terms of property extraction in the domain of conceptual knowledge. Baroni et al. extract features for concepts and the corresponding concept-feature relationships are described by way of 'type-sketches'. Only Devereux et al.'s method directly attacks the more difficult task of explicit, lexical relation extraction. Our aim is to expand and improve upon their methods, both in terms of the scope and accuracy of the acquisition as well as the thoroughness of evaluation. We have chosen the more ambitious goal of unconstrained property extraction because only this will properly emulate the desired property norm-like output. It also allows us to explore the capabilities and limitations of current NLP techniques for such a challenging task.

We will use Baroni et al.'s work as a principle point of comparison, and use De-

vereux et al.'s work as the starting point for our own research—it is more ambitious than Baroni et al.'s method, and thus more conducive to expansion and improvement. We hope also to harness ideas from the related literature in this work. There is plenty of scope for development: improving on the current pattern searching methods, experimenting with different corpora to see if qualitatively different relations/features can be extracted, introducing word association measures to further improve performance, enhancing the semantic feature analysis, improving evaluation methods (class-based and otherwise), incorporating relation-retrieval as an additional related task to the problem, experimenting with semi-supervised methods, and addressing issues of concept/relation/feature representation.

The implementation and development of sophisticated methods for the extraction of such property norm-like information necessitates a much 'deeper' understanding of the text. It requires a firm understanding of the linguistic phenomena which are typically indicative of the conceptually motivated properties we seek. The challenging nature of the task is compounded by the difficulties encountered when attempting to evaluate the system output. Few evaluation techniques for this task currently exist and, as mentioned already, the lists of properties gained from property norming studies are usually non-exhaustive and often inconsistent across concepts. As such there is no full "gold standard" for evaluating our work. We therefore also need to develop novel, accurate and conceptually motivated evaluation methods capable of addressing these problems. For these reasons we view our chosen topic to be appropriate and interesting from an NLP perspective: it focuses on a challenging real-world task in the cognitive sciences, and by pursuing it we will evaluate the extent to which existing NLP methodology can be both used and improved to address this task.

In the chapters that follow, we will explore issues of both methodology and evaluation that arise when attempting unconstrained, large-scale extraction of concept-relation-feature triples from corpus data.

Chapter 3

Pilot experiment

WE BEGIN OUR EXPERIMENTS using the system developed by Devereux et al. (2009) in collaboration with the *Computational Natural Language Processing and the Neuro-Cognition of Language Group* as an initial basic framework. As already mentioned, this system worked by analysing grammatical dependencies between concept words and associated feature-terms to extract potential concept-relation-feature triples from parsed data. It then used high level distributional information about co-occurring semantic concept- and feature-types to re-rank and filter the triples. We chose this system as it is sufficiently flexible to target all relation types and because we believe the underlying theory of the system to be sound and promising. However, we feel there is much room for improvement in the implementation, as well as plenty of scope for developing the method further by introducing more sophisticated rules and enhancing the filtering stage.

Since our task is unconstrained, and as such, poorly understood from an NLP perspective, this first experiment is investigative in nature: we will employ a rule- and knowledge-based method to create a system capable of broad extraction with high recall, but one which will allow for improving precision in later stages. In doing so, we hope also to obtain valuable information about our task that will guide the further development of our method using state of the art NLP techniques. Thus in this chapter, we improve on several components of the Devereux et al. system:

1. The method of basic property extraction from parsed data.
2. The distributional analysis of semantic concept and feature types.
3. The incorporation of the distributional analysis to guide property extraction.

Having made these improvements, we then assess the accuracy of the model using several different methods: a gold standard comparison, a qualitative analysis, fMRI

and EEG activation pattern prediction evaluations as well as a conceptual structure statistics evaluation.

This chapter contains work from two of our published papers *Acquiring human-like feature-based conceptual representations from corpora* (Kelly et al., 2010) and *Using fMRI activation to conceptual stimuli to evaluate methods for extracting conceptual representations from corpora* (Devereux et al., 2010).

3.1 Data

3.1.1 Recoded norms

As explained in Section 2.2, McRae et al. (2005) collected a set of norms listing properties for 541 concrete concepts. We use a modified, British English version of these norms in which concepts unfamiliar to British English speakers (e.g., ***gopher***, ***chickadee***) and superordinate concepts (e.g., ***toy***, ***building***) were removed, leaving 510 concepts for our experiments.[1] For the purposes of these experiments it was necessary to recode the relatively free-form McRae norms into a more coherent and structured representation to allow for easier and more rigorous computational manipulation, as well as to facilitate evaluation. Therefore, we recoded each property to a uniform ***concept*** *relation* **feature** representation (e.g., ***bat*** *have* **wing**) suitable for our experiments. This was done as follows: if the property was a behaviour, then the infinitive of the verb acted as the feature and the verb *do* acted as the relation. Otherwise, the final noun/adjective of the property was employed as the feature and the main (non-auxiliary) verb was used as the relation. We removed all determiners: the vast majority of the norms contain general properties, with very few containing determiners whose specificity is likely to significantly alter a given property's meaning. We also removed prepositions: although we accept that there is a sizeable minority of the norms—typically functional properties—for which prepositions can carry meaning (e.g., the relations *used for*, *used by*, *used with* and *used in* are clearly semantically different), we decided that, at this early stage, evaluating our system without prepositions would offer a better overall picture of its performance. Next, if the property contained an adjective-noun combination this would be recoded and split into two separate concept-relation-feature triples. Although this separation of concept properties into constituent key sections (and omission of certain aspects of some of the original properties) is a simplification, the amount of information lost is actually relatively small—in the vast majority of cases, the three-term triples returned are true to their original meaning. Table 3.1 gives the ten most

[1] See Taylor et al. (2011) for details.

frequent (normalised) features for two concepts in the norms, ***car*** and ***penguin***, and their corresponding recoded triples—it is triples of this form that we aim to extract.

car

McRae property	Recoded triple	Freq.
has wheels	*have* **wheel**	19
used for transportation	*use* **transportation**	19
has 4 wheels	*have* **4-wheel**	18
has doors	*have* **door**	13
has an engine	*have* **engine**	13
requires petrol	*require* **petrol**	12
has a steering wheel	*have* **steering-wheel**	12
used for passengers	*use* **passenger**	9
a vehicle	*be* **vehicle**	9
is fast	*be* **fast**	9

penguin

McRae feature	Recoded triple	Freq.
is black	*be* **black**	24
a bird	*be* **bird**	22
is black and white	*be* **black-and-white**	22
is white	*be* **white**	22
has a beak	*have* **beak**	21
beh—cannot fly	*cannot* **fly**	20
beh—waddles	*do* **waddle**	15
beh—swims	*do* **swim**	14
lives in cold climates	*live* **climate**	13
has feet	*have* **foot**	12

Table 3.1: Sample properties from the McRae norms for ***car*** and ***penguin*** with their citation frequencies and the corresponding recoded concept-relation-feature triples.

3.1.2 Corpora

We employ three corpora for our experiments: two are subsets of Wikipedia (the Wiki500 and Wiki100K corpora), and the other is the British National Corpus (BNC) (Leech, 1992).

We chose Wikipedia because it forms a large, comprehensive and freely accessible source of encylopedic knowledge and we are confident that much of the property norm information we seek is likely to be encoded within it. Indeed, nearly all the McRae concepts have their own articles in Wikipedia, and generally the majority of cited properties for a given concept can be found in its Wikipedia article; the articles

often include facts similar to those elicited in norming studies,[2] albeit rarely expressed in an identical (or indeed similar) way to the property norms.

The 1.84 million articles from Wikipedia were compiled into two subcorpora. The Wiki500 corpus (1.1 million words) contains around 500 Wikipedia articles corresponding to each of the McRae concepts, whilst the Wiki100K corpus (36.5 million words) comprises those Wikipedia articles with titles containing one of the McRae concepts (and with a title-length of five words or less).[3] Extraneous data were removed from the articles (e.g., infoboxes, bibliographies) to create a plaintext version of each article. This corpus, which we call Wiki100K, holds 109,648 plaintext articles (36.5 million words). For a full description of how the Wikipedia subcorpora were generated, see Devereux et al. (2009).

The 100-million word BNC contains written (90%) and spoken (10%) UK English collected from 1960 to 1993. It is balanced across domains in that it is not limited to any particular subject, genre or field and is designed to represent a broad cross-section of modern British English.

These corpora were chosen to illustrate the extent to which the type of property norm-like information we seek can be found in both encyclopedic and general contexts. Although we might expect much of the human-produced knowledge found in the McRae norms to also exist in this encylopedic resource—implying a certain degree of completeness to Wikipedia with regard to our task—this is not always the case. For example, the triple *eaten by* **monkeys** appears as a property of ***banana*** in the McRae norms, but the word 'monkey' does not appear at all in the Wikipedia *banana* article. Hence, we also hope to assess the extent to which those properties not included in Wikipedia (perhaps due to their incidental rather than scientific nature, or ambiguity for encyclopedic purposes) might instead be encoded in everyday speech and text, such as that contained in the BNC. We assume that any properties contained in such a wide-ranging corpus would be presented implicitly, rather than explicitly stated. We will later compare the properties returned from each corpus to investigate whether they complement one another, as this would motivate using a combination of the two.

3.1.3 Parser

We parsed the corpora using the Robust Accurate Statistical Parsing (RASP) system (Briscoe, 2006). For each sentence in each corpus this yields the most probable analysis returned by the parser in the form of a set of grammatical relations (GRs), or, should

[2]For example, the article *elephant* describes how elephants are large, are mammals, and live in Africa.

[3]This was done in order to avoid articles on very specific topics which are unlikely to contain basic information about the target concept.

the parse fail, the GRs for the most likely sequence of subanalyses. The generated GRs are head-based dependencies (Carroll and Briscoe, 2002) which follow the format:

```
(<GR-type> <optional-subtype> <head>
    <dependent> <optional-initial-GR>)
```

The following example from Briscoe (2006) illustrates how GRs are used to represent grammatical dependencies (indirect object, direct object, etc.) between a head and its dependent.[4] The sentence

Kim flew to Paris from Geneva.

would be parsed as

```
(ncsubj flew Kim _) (iobj flew to) (iobj flew from)
   (dobj to Paris) (dobj from Geneva)
```

The RASP parser also produces part of speech (POS) information for each word in the sentence.

Much of the recent work on this task has used only shallow techniques (e.g., part of speech information and word/sentence windows) to extract the types of information we seek (e.g., the Strudel system by Baroni et al. (2009)). However computational methods for extracting semantic information from text often perform better when taking syntactic information into account (Clark and Weir, 2002; Padó and Lapata, 2007). Indeed, Baroni and Lenci (2010) use parsed text to instantiate their 'Distributional Memory' framework which is designed to act as a generalised distributional model of language suitable for a number of NLP tasks including property extraction. Following from the intuition that entities which have a relationship in the world will likely also be grammatically linked in sentences containing those entities, using the GR output offers us the possibility to analyse the underlying structure of the terms within a sentence and predict meaningful relationships based on that structure. We believe this is preferable to simply relying on co-occurrence strength and part of speech information to surmise relationships.

3.2 Method

Our method for extracting concept-relation-feature triples consists of two main stages. In the first stage, we extract large sets of candidate concept-relation-feature triples for

[4]For more detail on RASP GR output, see Briscoe (2006).

each target concept from parsed corpus data. In the second stage, we rerank and filter these triples with the intention of retaining only those triples likely to be true semantic properties.

3.2.1 Manual rule construction

In the first stage, the GR sets for each sentence containing a target concept noun are retrieved from the corpus. From this we derive a directed acyclic[5] graph (DAG) where the nodes are labelled with words in the sentence and their parts of speech, and the edges with the grammatical relations linking the nodes together. Using this DAG we can then easily generate all possible paths which are rooted at the target concept node using a breadth-first search.[6]

We then examine whether any of these paths match prototypical feature-relation GR structures according to our manually generated rules. The rules were created by first extracting features from the McRae norms for a small subset of the concepts and extracting those sentences from the Wiki500 corpus which contained both concept and feature terms. For each sentence, we examined the intermediate terms along each path through the graph (containing the GRs and POS tags) linking the concept and the feature and—providing no other rule would already generate the concept-relation-feature triple—manually generated a rule based on each path.

For example, the sentence

There are also aprons that will cover the sleeves.

should yield the triple ***apron*** *cover* **sleeve**. We examine the graph structure of the sentence rooted at the concept ***apron***, as shown in Figure 3.1.

Here, the relation is relatively simple—we merely create a rule requiring that the relation be a verb (i.e., has a `V` POS tag), the feature has a `NN` tag and that there is a `dobj` GR linking the feature to the concept. Our rules are effectively a constraint on (a) which paths should be followed through the graph, and (b) which items in that path should be noted in the concept-relation-feature triple. By creating several such rules and applying them to a large number of sentences, we extract potential features and relations for the concepts.

We avoided specifying too many POS tags and GRs in our rules, since this could have resulted in too few matching paths. In the above rule, we could have required also

[5]It should be noted that RASP grammatical relation output is *almost* acyclic: Briscoe (2006) states that "The aim is that the GR scheme is a factored representation of a directed graph which is almost acyclic." For those times when the graph is cyclic we process the relations in order of appearance in the parse and ignore any subsequent edges which would cause cycles.

[6]In constructing this DAG, we ignore the directionality associated with the GRs.

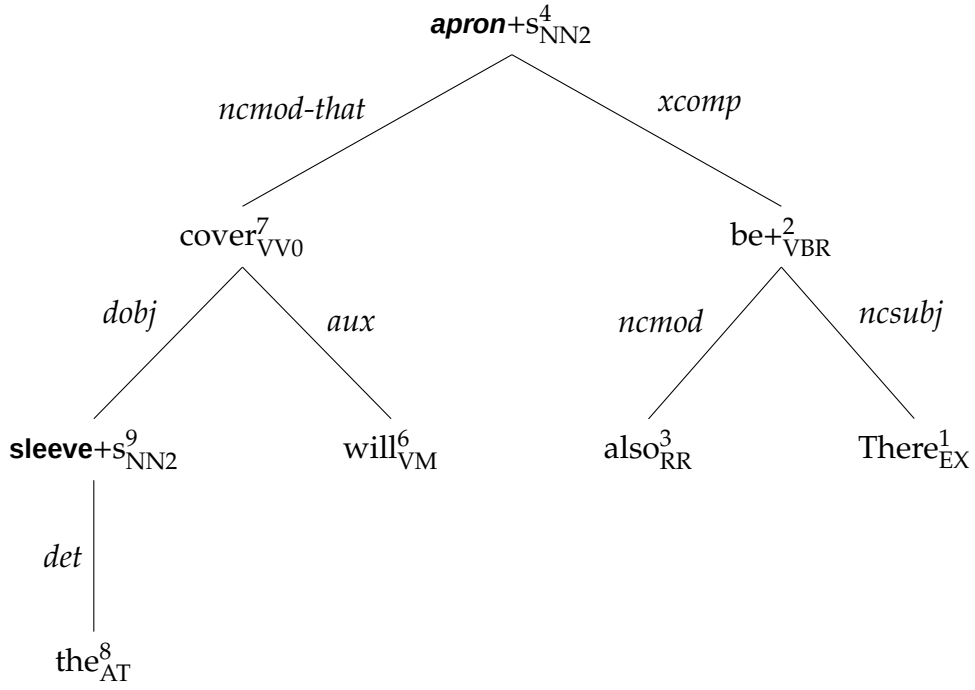

Figure 3.1: A RASP-derived GR-POS graph for the sentence *There are also aprons that will cover the sleeves.*

a `cmod-that` relation linked the feature and concept—but this would have excluded sentences like *the apron covered the sleeves*. Conversely, we avoided making our rules too permissive. For example, eliminating the `dobj` requirement from the above rule would have yielded the triple ***apron*** *be* **steel** from the sentence *the apron hooks were steel.*

We applied this manual rule-construction process iteratively. That is to say, we generated a small number of rules, applied them to the sentences and examined the output triples and the non-activated sentences. Rules with a tendency towards producing plausible triples were retained, whilst those which mostly produced incorrect triples were discarded. We repeated this process until we were left only with sentences with overly specific rule-prototypes[7] and sentences containing false feature-relation matches.[8]

Following this process on the Wiki500 corpus for around a dozen concepts eventually produced fifteen rules, listed in Table 3.2 (with example sentences and examples of produced concept-feature-relation triples). It is noteworthy that although we always use the feature found by the rule as the feature of the triple, the relation is not neces-

[7] e.g., sentences such as "Inside the accordion are the reeds that generate the instrument tones"—we could be extracting *accordion has reeds* from this sentence, but we believe the *inside the* ***x*** *are* ***y*** pattern is overly specific and wouldn't always yield a *has* relation.

[8] e.g., sentences such as "the tiger was next to the building with stripes".

sarily contained in the source sentence (e.g., Rules 4, 8, 9 and Rules 11–15)—the benefit of doing this is that it enables us to harness information that is only implicit from the text and make it explicit (for example Rule 8 means we are able to extract *tiger has tail* from a phrase such as *the tiger's tail was long*).

Table 3.2: Our 15 rules, with frequency information of rule-firing on the Wiki500 corpus, the rule itself, an example sentence and the resulting output triple found from applying the rule to the sentence.

ID	Freq.	Rule	Sentence	Triple
1	2,174	relation has a `VVN` tag and feature has a `NN` tag and they are linked by a `xcomp` GR	This is an anchor which relies solely on being a heavy weight.	***anchor*** *be* **weight**
2	449	Relation is a verb and feature is an adjective and they are linked by a `xcomp` GR.	Sliced apples turn brown with exposure to air due to the conversion of natural phenolic substances into melanin upon exposure to oxygen.	***apple*** *turn* **brown**
3	2,002	Feature has a `VV0` tag and relation is a verb and they are linked by a `aux` GR.	Grassy bottoms may be good, but only if the anchor can penetrate the foliage.	***anchor*** *can* **penetrate**
4	4,378	Feature tag contains `N` which is linked to its parent node by a `iobj` GR, which in turn is linked to its parent node by a `dobj` GR and the path from concept to feature only contains `det`, `xmod`, `iobj` and `dobj` GRs. The relation is saved as *has*.	The Chilean version of caesar salad contains large slices of mature avocado.	***avocado*** *have* **slices**
5	42,355	The feature has a `NN` tag and there is only one verb in the path from concept to feature. The relation is saved as this verb.	The axe symbolized the authority to execute and were often used as symbols for Fascist Italy under Mussolini.	***axe*** *symbolise* **authority**
6	4,194	Relation is a verb and feature has a `NN` tag and they are linked by a `dobj` GR.	There are also aprons that will cover the sleeves.	***apron*** *cover* **sleeves**
7	258	The relation is a verb and the feature is linked to the previous node by a `obj` GR.	By the Oligocene and Miocene ants had come to represent 20-40 % of all insects found in major fossil deposits.	***ant*** *represent* **%**

Table 3.2: (continued)

ID	Freq.	Rule	Sentence	Triple
8	502	The path from concept to feature is of length 1 or 2 and the final edge of that path is a `ncmod-poss` GR. The relation is saved as *has*.	Alligators' main prey are smaller animals that they can kill and eat with a single bite.	***alligator*** *have* **prey**
9	7,332	The relation has a `IO` tag and the feature has a `NN` tag and they are linked by a `iobj` GR. The relation is saved as *has*.	Armour often also bore an insignia of the maker, especially if it was of good quality.	***armour*** *have* **quality**
10	1,016	The path length is greater than 3 and the feature has a `NN` tag and it is linked to the previous node by a `ncmod` relation and the path from concept to feature only contains `ncmod`, `conj` and `ta-bal` GRs. The relation is saved as *has*.	Cairo in the 16th century had high-rise apartment buildings where the two lower floors were for commercial and storage purposes and the multiple stories above them were rented out to tenants.	***apartment*** *have* **storage**
11	7,452	The path from concept to feature is of length 1 or 2 and the feature has a `NN` tag and the final edge of that path is a `ncmod` GR. The relation is saved as *has*.	The axe blade is 17,4 cm long and made of antigorite, mined in the Gotthard-area.	***axe*** *have* **blade**
12	1,631	The feature has a `VV0` tag and the feature is linked to its previous node by a `conj` GR. The relation is saved as *can*.	Research is also ongoing in electromagnetic armour systems to disperse or deflect incoming shaped charge jets.	***armour*** *can* **deflect**
13	6,628	The feature has a `JJ` tag and the path from concept to feature only contains `ncmod` and `conj` relations. The relation is saved as *is*.	Premium supermarkets sell pre-ripened avocados treated with synthetic ethylene to hasten the ripening process.	***avocado*** *be* **pre-ripened**

Table 3.2: (continued)

ID	Freq.	Rule	Sentence	Triple
14	2,183	The feature has a VVN tag and the feature is linked to the previous node by a conj GR. The relation is saved as *is*.	Historically avocados had a long-standing stigma as a sexual stimulant and were not purchased or consumed by any person wishing to preserve a chaste image.	***avocado*** *be* **consumed**
15	2,045	The path length is 4 and the feature is linked to the previous node by a conj GR and the feature has a NN tag. The relation is saved as *has*.	Some modern day aprons will have humorous expressions, designs or corporate logos.	***apron*** *have* **logos**

3.2.2 Improving the basic property extraction

The second stage of our method evaluates the quality of the extracted candidate triples generated in the first stage using semantic information, with the aim of filtering out poor quality properties. We would expect the number of times a triple is extracted for a given concept to be proportional to the likelihood that the triple represents a true property of that concept. However, production frequency alone is not a sufficient indicator of quality, because concept terms can produce unexpected candidate properties.[9]

We attempt to address this issue by, as Devereux et al. (2009) did, introducing the notion of semantic categories. In other words, the probability of a property being part of a concept's representation is dependent on the semantic category to which the concept belongs (for example, *used for* **cutting** would be expected to have low probability for animal concepts). We analysed the norms to quantify this type of semantic information with the aim of identifying higher-order structure in the distribution of semantic classes for features and concepts. Our goal is to determine whether this information can indeed improve the accuracy of property extraction.

We assume that there is a probability distribution over concept and feature classes, $P(F|C)$, where C is a concept class (e.g., *Apparel*, *Instruments*) and F is a feature class (e.g., *Materials*, *Activities*). Knowing this distribution provides us with a means of assessing how likely it is that a candidate feature $f \in F$ is relevant to a concept $c \in C$ by using their membership in their respective classes to extract the conditional probability $P(F|C)$. The McRae norms may be considered to be a sample drawn from this distribution if the concept and feature terms appearing in the norms can be assigned to suitable concept and feature classes. These classes were identified by way of clustering.

Semantic similarity

The first step required for clustering involves establishing how similar terms are to one another. WordNet (Fellbaum, 1998) is a large lexical database of English, where words are grouped into synonym sets, each corresponding to a different concept, and the semantic relations between those sets are recorded. Various algorithms have been proposed for using the WordNet hierarchy to establish the similarity of two terms, usually based on some definition of proximity within WordNet's lexical ontology. For these experiments, we employ the Lin (1998) similarity measure. For example, ***lion*** and ***cat*** have a high Lin similarity (0.8509) in WordNet, while ***lion*** and ***axe*** have a much lower similarity (0.1430).

[9] For example, one of the extracted triples for ***tiger*** is ***tiger*** *have* **squadron** because of the RAF squadron called the Tigers.

This metric is suitable for our task since we would like to generate appropriate superordinate classes for which we can calculate distributional statistics. We could merely cluster on the most frequent sense of concept and feature words in WordNet, but the most frequent sense in WordNet may not correspond to the intended sense in the property norm data. For example, the first and second most frequent definitions of ***kite*** in WordNet refer to a slang meaning for the word ***cheque***—only the third most frequent meaning refers to ***kite*** as a toy, which most people would understand to be its predominant sense. We therefore experimented with three distinct WordNet sense-choice functions to calculate the similarity between two words, w_1 and w_2:

- **Mostfreq**. Chooses the most frequent sense of w_1 and w_2 respectively in WordNet and calculates the similarity between those two senses.
- **All**. Chooses the senses s_{1i} and s_{2j} of w_1 and w_2 respectively such that $\text{sim}(s_{1i}, s_{2j})$ is maximised. This strategy has been employed elsewhere (Resnik, 1995).
- **Manual**. Employs a manually annotated list to choose the correct sense in WordNet.[10] This is only possible for concept clustering as we don't have a manual WordNet sense annotation for the 7000 McRae features.

Our initial analysis indicated that the Manual method worked the best for concept clustering, and for the feature clustering we backed off to the Mostfreq method.

Clustering techniques

We wanted to explore the impact of using different clustering methods for these experiments. To this end, we clustered concepts and feature terms appearing in the recoded norms independently into 50 clusters using three methods: hierarchical clustering, *k*-medoids clustering and non-negative matrix factorisation (NMF). We chose hierarchical clustering because it could potentially harness the relatively hierarchical nature of noun concepts and features, especially in WordNet; *k*-medoids because it produces size-balanced clusters with few very small (size 1 or 2) clusters; and NMF because it has performed well in other word-clustering tasks (e.g., document clustering (Xu et al., 2003)).

- **Hierarchical clustering**. Agglomerative hierarchical clustering works by considering each individual element to be its own cluster, and at each iteration merging the two clusters which are 'closest' to each other. This is done until a threshold (e.g., a minimum number of clusters, or a maximum intra-cluster distance)

[10] Thanks to Barry Devereux for performing this annotation.

***k*-medoids**		
banjo	biscuit	blackbird
bat	cup	ox
beehive	kettle	peacock
birch	sailboat	prawn
bookcase	shoe	prune

NMF		
ashtray	bouquet	eel
bayonet	cabinet	grapefruit
cape	card	guppy
cat	cellar	moose
catfish	chandelier	otter

Hierarchical		
Fruit/Veg	*Apparel*	*Instruments*
apple	apron	accordion
avocado	armour	bagpipes
banana	belt	banjo
beehive	blouse	cello
blueberry	boot	clarinet

Table 3.3: First five elements alphabetically from three sample clusters for the three clustering methods.

is reached. There are various methods used for deciding the distance between A and B, dist(A, B). These methods include complete-linkage (where dist(A, B) is defined as the distance between the two elements of A and B which are furthest from one another), single-linkage (where dist(A, B) is defined as the distance between the two elements of A and B which are closest to one another) and average-linkage (where dist(A, B) is the mean of dist(a, b) for all $(a, b) \in A \times B$). For these experiments we used average-linkage, and a threshold of 50 clusters.

- ***k*-medoids clustering**. We also use a partitional clustering algorithm known as *k*-medoids (a descendant of *k*-means clustering). A medoid of a set is defined to be a data point in that set with minimal average dissimilarity to all other data points in that set. The *k*-medoids algorithm works by initially randomly assigning k of the elements to be the medoids. The remaining points are then assigned to their closest cluster. Then for each medoid m, the algorithm examines each element e in m's cluster and calculates the cost (in terms of increasing/decreasing overall distance between clusters) of swapping e and m. It selects e with the lowest cost

	Fruit/Veg	Apparel	Instruments
Plant Parts	0.144	0.037	0.008
Materials	0.006	0.148	0.008
Activities	0.009	0.074	0.161

Table 3.4: $P(F|C)$ for $C \in$ {Fruit/Veg, Apparel, Instruments} and $F \in$ {Plant Parts, Materials, Activities} when using hierarchical clustering.

calculation and swaps e with m. This process is repeated until there is no change in m.

- **Non-negative matrix factorisation**. NMF works by taking a matrix X (our similarity matrix) and factorising it[11] into two non-negative matrices U and V, such that $X \approx UV$. We can view the elements $u_{ij} \in U$ and $v_{ij} \in V$ as representing the degree to which the ith term (t_i) belongs to cluster j. To combine these variables, we follow the method presented by Xu et al. (2003) to normalise both matrices so that the Euclidean lengths of the column vectors of matrix U are 1, whilst ensuring $UV = U'V'^T$. We may then use the matrix V' to determine a cluster label for each term t_i where t_i is assigned to cluster c if $c = \arg\max_j v'_{ij}$.

We show the first five alphabetical elements from three of the clusters produced by the three clustering methods in Table 3.3. The hierarchical clustering appears to be producing the most intuitive clusters.

Estimating probability distribution

We clustered (using all three techniques) both the concepts and the features using production frequency information from the McRae norms to estimate the probability distribution, $P(F|C)$, over all concept clusters, C, and feature clusters, F:

$$P(F|C) = \frac{P(C,F)}{P(C)} = \sum_{c \in C, f \in F} \text{freq}(c,f) \Big/ \sum_{c \in C} \text{freq}(c) \tag{3.1}$$

Then, for an individual triple, to derive the conditional probability of f given the concept c we merely use $P(F|C)$ where C and F are chosen such that $c \in C$ and $f \in F$. We call this conditional probability for a given triple our semantic reweighting factor.

For example, Table 3.4 shows a sample of $P(F|C)$ values for three concept classes and three feature classes and Table 3.5 shows example members of the same three feature classes when using hierarchical clustering. We can see that $P(\textit{Materials}|\textit{Apparel})$

[11] To find this factorisation, we use a modified version of Lin's projected gradients implementation (2007) with randomised initial matrices.

Hierarchical Clustering		
Plant Parts	*Materials*	*Activities*
berry	cotton	annoying
bush	fibre	listening
core	nylon	music
plant	silk	showing
seed	spandex	looking

Table 3.5: Example members of feature clusters for hierarchical clustering.

is higher than $P(Materials|Fruit/Veg)$: given a concept in the *Apparel* cluster, the probability of a *Materials* feature is relatively high, whereas given a concept in the *Fruit/Veg* cluster, the probability of a *Materials* feature is low. This cluster analysis therefore supports our hypothesis that the likelihood of a particular feature for a particular concept is dependent on the semantic categories to which both belong.

3.2.3 Reranking

We investigated whether this distributional semantic information could be used to improve the quality of the candidate triples by using the conditional probabilities of the appropriate feature cluster given the concept cluster as a weighting factor. To obtain the probabilities for a triple, we first find the clusters to which the concept and features words belong. If the feature word of the extracted triple appears in the norms, its cluster membership is drawn directly from there; if not, we assign the feature to the feature cluster with which it has the highest average similarity.[12] Having determined the concept and feature clusters for the triple, we reweight its raw corpus occurrence frequency by multiplying it by the calculated conditional probability. In this way, incorrect triples that occur frequently in the data are downweighted, and more plausible triples have their ranking boosted.

3.3 Evaluation

There are a number of potential methods for evaluating the quality of the extracted triples. One possibility would be to calculate standard NLP measures of precision and recall for the extracted triples with respect to the McRae norms 'gold standard'. However, direct comparison with the recoded norms is difficult, since there may be

[12]We use average-linkage for hierarchical and k-medoids clustering, and mean cosine similarity for NMF.

extracted features which are semantically equivalent to a triple found in the norms but possessing a different lexical form. For example, ***avocado*** *have* **stone** appears in the recoded norms whilst our method extracts ***avocado*** *contain* **pit**; direct comparison of these two triples therefore incorrectly judges ***avocado*** *contain* **pit** to be incorrect. Similarly, property norms are typically normalised so that near-synonymous features (e.g., **water**, **sea**, **ocean** for the concept ***whale***) given by different participants are mapped to the same feature label (e.g., **water**). As a consequence, a model may correctly extract *lives in* **sea** for ***whale***, but **sea** will not match any feature in the norms if all such properties are normalised to *lives in* **water**. We therefore need to employ more sophisticated and additional evaluation techniques capable of addressing these problems.

To reduce the impact of these issues, we followed the approach taken in the *ESSLLI Distributional Lexical Semantics Workshop 2008* as our gold standard. To evaluate the ability of our model to generate novel properties, we will also conduct a manual evaluation of the highest-ranked extracted triples that did not appear in the norms. We also use Mitchell et al.'s fMRI activation data (Mitchell et al., 2008) to attempt to predict which of two concepts an fMRI image corresponds to. Murphy et al. (2009) similarly offer EEG activation data from a silent naming task for a number of concrete concepts which we will use to similarly predict which of two concepts a specific set of activation data corresponds to. Finally, we employ a conceptual structure statistics evaluation to compare the structural properties of the output with that of the McRae norms.

3.3.1 SVD baseline

We use a baseline to compare the performance of our approach to a non-task-specific, co-occurrence based technique. To this end, we use the 'SVD' baseline[13] as described by Baroni et al. (Baroni and Lenci, 2008; Baroni et al., 2009). It combines aspects of the HAL (Deerwester et al., 1990) and LSA (Lund and Burgess, 1996) models and is a simple word association method, not tailored to extracting properties.

To construct the baseline method we begin by defining 'context words' to be the 5000 most frequent content words (i.e., excluding those from a stoplist) in the lemmatised corpus, and 'target words' as the concept terms in the McRae norms supplemented with the 10,000 most frequent content words in the corpus (excluding the ten most frequent words). The model works by creating a co-occurrence matrix, summing how often each target word co-occurs with each context word within a sentence over all sentences in the corpus. The context window was defined as within sentence boundaries because this is analogous to our experimental rule extraction method.

[13]Thanks to Barry Devereux for his implementation of the 'SVD' baseline, which was only slightly modified for this work.

The dimensionality of this target-word/context-word co-ocurrence matrix was then reduced to 150 columns by singular value decomposition, and similarity between pairs of target words was calculated as the cosine between their respective columns. Then for each concept word the 200 closest target words were considered as the extracted feature terms for that concept.

3.3.2 Gold standard evaluation

In NLP, it is typical to evaluate performance by comparing system output with a gold standard such as (a subset of) the McRae norms, using classic NLP precision, recall and F-scores. The precision score for a given concept's properties is defined as the size of the overlap between the correct properties for that concept (where a property is defined as 'correct' if it exists in the McRae norms) and the extracted properties for that concept, divided by the total number of extracted properties (i.e., the fraction of retrieved properties which are relevant to the concept). Recall is defined as the size of the overlap between correct properties and extracted properties divided by the total number of correct properties (i.e., the fraction of relevant properties which are successfully retrieved). The F-score is the harmonic mean of precision and recall.

Employing these measures directly on the McRae norms presents additional difficulties in a true evaluation of system output to those discussed earlier, since lexically different statements often represent semantically identical properties (e.g., ***jar*** *used for* **jelly** and ***jar*** *used for* **jam**). Therefore we will employ the gold standard used at the *ESSLLI Lexical Semantics Workshop 2008*, "Task 3: Generation of salient properties for concepts" (Baroni et al., 2008). There they created a new gold standard which comprised the top ten lemmatised properties for each of 44 concepts from the recoded McRae norms (the concepts belonged to 6 semantic categories: four animate and two inanimate).

In addition to the ten features for each concept, an 'expansion set' was generated for each concept-feature pair. This set was built to tackle two problems which arise when determining whether a property generated by a model matches a property in the McRae norms. First, all current models produce single words as properties, and these have to be matched against multi-word phrases in the norms. Second, McRae et al. normalised their properties by channelling synonymous properties into a single representation. It is for these reasons that we try not to compare directly against the original McRae norms. In other words, the ESSLLI expansion set attempts to undo the normalisation process described in Section 2.2 so that e.g., **loud**, **noise** and **noisy** can all be counted as matches against the property *is* **loud**.

This expansion set was constructed by first extracting from WordNet the synonyms

of the words that constituted the concepts' features, then manually filtering out irrelevant synonyms and finally inserting other potential matches, including inflectional and derivational variants (**leg** for **legs** and **transport** for **transportation**), as well as other semantic neighbours or closely related entities. For example, the property *lives on* **water** was expanded to the set {*lives on* **aquatic**, *lives on* **lake**, *lives on* **ocean**, *lives on* **river**, *lives on* **sea**, *lives on* **water**}. This expansion relied on somewhat subjective human judgements. However, we, like the workshop authors, believe it offers a better evaluation than comparing directly against the McRae norms, as it allows comparison of extracted property labels to the gold standard without insisting on exact lexical matching.

Although the ESSLLI set only contains properties for a subset of the McRae concepts, since the chosen concepts are diverse in terms of their semantic categories, we believe they represent a fair and reasonably comprehensive benchmark against which to evaluate. Furthermore, the ESSLLI norms provide a close match to the target norms—by using them we are enforcing the most strict evaluation possible without falling foul of the problem of misaligned synonyms between our gold standard and extracted properties. We will compare the generated triples with this expansion set using precision, recall and F-measures.

We note, however, that this set does not include expansions of relation labels. McRae (2012) observed that "there was a great deal of variance" in the responses offered for the norming study, and the final choices of relations found in the McRae norms was the result of number of contributing factors: some were plainly obvious, while others had historical precedent (e.g., *a* for the *is a* relationship) or had been chosen to illustrate certain semantic distinctions (e.g., ***car*** *has* **wheels** vs ***car*** *requires* **driver**, rather than ***car*** *has* **driver**—the relation *requires* used because **driver** is not a 'part' of a car).

Furthermore, there are often multiple ways to express the relationship between a concept and its feature. For example, consider the phrases "cars have doors", "the car doors", "she opened the door and got into her car", "this car's a three-door". We are hoping to funnel all of these expressions into the ***concept*** *relation* **feature** pattern, yet we can see that the desired *relation* portion (in this case *has* of ***car*** *has* **doors**) may not explicitly appear in the sentences we are extracting from. There is also little semantic difference between, for example, ***car*** *has* **doors**, ***car*** *includes* **doors**, yet we will only consider one of these to be correct when comparing directly against the McRae norms. Unfortunately we do not have access to a synonym-expanded set of relation labels to compensate for the above issues and therefore when comparing the extracted relations directly with the norms we should bear in mind that this constitutes a tough standard of evaluation. Indeed, previous large-scale models of property extraction have been evaluated on concept-feature pairs rather than triples (e.g., Baroni et al., 2009).

We will aim to attain high recall when evaluating against the ESSLLI set (since, ideally, all properties in the norms should be extracted). We are somewhat less concerned about achieving very high precision as we still expect to find correct properties in corpus data that do not appear in our gold standard: extracted properties that are not in the norms may still be correct (e.g., *breathes* **air** for ***tiger***).

Table 3.6 presents the results of our method when we evaluate using the feature term alone (i.e., in calculating precision and recall, we disregard the relation verb, and require only a match between the feature terms in the extracted triples and those in the recoded norms). Results for six sets of extractions are presented. The first set is the set of features extracted by the SVD baseline. The second set consists of the full set of triples extracted by our method, prior to the reweighting stage. 'Top 20 unweighted' gives the results when all but the top twenty most frequently extracted triples for each concept are filtered out. Note that the filtering criterion here is raw extraction frequency, without reweighting by conditional probabilities. 'Top 20 (*clustering type*)' are the corresponding results when the features are weighted by the conditional probability factors (derived from the three clustering methods) prior to filtering; that is, using the top twenty reranked features. The effectiveness of using the semantic class-based analysis data in this method can be assessed by comparing the filtered results both with and without feature weighting.

For the baseline implementation, the results are better when we use the smaller Wiki500 corpus compared to the larger Wiki100K corpus. This is perhaps not surprising, since the smaller corpus contains only those articles corresponding to the concepts found in the norms. This smaller corpus seems to minimise noise due to phenomena such as word polysemy, which would be more apparent in larger corpora.

The results for the baseline model and the unfiltered method are quite similar for the Wiki500 corpus, whilst the results for the unfiltered method using the Wiki100K corpus give the maximum recall achieved by our method: 89.4% of the features are extracted, although this figure is closely followed by that of the BNC at 88.1%. As the unfiltered method was constructed to overgenerate, a large number of features are being extracted and therefore precision is low.

For the results of the filtered method, where all but the top twenty triples were discarded, we can see the benefit of reranking. The reranked output for all three clustering types yields much higher precision and recall scores than the unweighted method. Our best performance is achieved using the BNC and hierarchical clustering, where we obtain 19.4% precision and 38.9% recall. It is clear therefore that both general and encyclopedic corpus data can prove useful for the task. An interesting question is whether these two data types offer different, complementary feature types for the task.

Extraction set	Corpus	Prec.	Recall	F
SVD Baseline	Wiki500	0.0235	0.4712	0.0448
	Wiki100K	0.0140	0.2798	0.0266
	BNC	0.0131	0.2621	0.0249
Unfiltered	Wiki500	0.0242	0.6515	0.0467
	Wiki100K	0.0039	0.8944	0.0077
	BNC	0.0042	0.8813	0.0083
Top 20 (unweighted)	Wiki500	0.1159	0.2326	0.1547
	Wiki100K	0.0761	0.1523	0.1015
	BNC	0.0841	0.1692	0.1123
Top 20 (hierarchical clustering)	Wiki500	0.1693	0.3394	0.2259
	Wiki100K	0.1733	0.3553	0.2365
	BNC	0.1943	0.3896	0.2593
Top 20 (*k*-medoids clustering)	Wiki500	0.1159	0.2323	0.1547
	Wiki100K	0.1000	0.2008	0.1335
	BNC	0.1216	0.2442	0.1623
Top 20 (NMF clustering)	Wiki500	0.1375	0.2755	0.1834
	Wiki100K	0.1409	0.2826	0.1880
	BNC	0.1500	0.3010	0.2002

Table 3.6: Precision, Recall and F-scores for our pilot system when matching on features only.

Extraction set	Corpus	Prec.	Recall	F
Top 20 (hierarchical clustering)	Wiki500	0.1011	0.2028	0.1349
	Wiki100K	0.1102	0.2210	0.1471
	BNC	0.0955	0.1917	0.1275

Table 3.7: Precision, Recall and F-scores for our best method when matching on features and relations.

We discuss this point further in the next section.

Using exactly the same gold standard, Baroni et al. (2009) obtained precision of 23.9% when extracting the top ten features. On the same evaluation criteria and using the BNC corpus, our system achieves a best precision of 29.6%. Devereux et al. (2009) achieved a best precision score of 6.5% on the same test set, however this is again not directly comparable with the results above because they elected to evaluate the top 25% of their returned properties. Our best result on the top 25% of features returned is 8.0% on the Wiki500 corpus—its smaller size means that fewer features are returned compared to the other two corpora.

One of the innovations of our method is that it uses information about the GR-graph of the sentence to also extract the relation appearing in the path linking the concept and feature terms in the sentence (or predicts the relation, based on this path). This is not possible in a purely co-occurrence-based model. We therefore also evaluated

the extracted triples using the full relation and feature pair (i.e., both the feature and the relation verb needed to match the gold standard). The results for our best method are shown in Table 3.7. Unsurprisingly, because this task is more difficult, precision and recall are reduced. However, since we enforce no constraints on what the relation may be, other than matching one of our relatively general rules, and since we do not have expanded synonym sets for the relations (as we do for the features), we think it is actually quite remarkable to have the relation verb and feature exactly matching with the recoded norms almost one in every five times.

3.3.3 Qualitative analysis

We now examine the qualitative differences in extraction between the encyclopedic and general corpora. When considering the top twenty features extracted using our best method applied to the Wiki500 corpus versus the BNC corpus, the overlap of features is relatively low at 22.73%. When one also takes the extracted relations into account, this figure reduces to 6.45%. It is therefore clear that relatively distinct groups of features are being extracted from the encyclopedic and general corpus data. This could motivate combining the corpora for improved performance: we will come back to this later.

To further illustrate the nature of these differences between the types of properties being extracted, we use our best method as described in the previous section to show the top ten extracted distinct properties for three concepts (***swan***, ***pineapple*** and ***screwdriver***) from the Wiki500 corpus and the BNC corpus in Table 3.8. We label those properties that are correct according to the norms as Correct (C), those which do not appear in the norms but we believe to be plausible as Plausible (P), and those that do not appear in the norms and are also implausible as Incorrect (I). This analysis revealed that many of the errors were not true errors, but potentially valid triples missing from the gold standard. We can see that our method has detected several plausible triples not appearing in the norms (and consequently, our gold standard), e.g., ***swan*** *have* **chick** and ***screwdriver*** *be* **sharp**. It should also be pointed out that some 'incorrect' properties (e.g., ***screwdriver*** *achieve* **goal**) could be considered to be at least broadly accurate. We recognise that the ideal evaluation for our method would involve having human participants assess the extracted properties for a diverse cross-section of the concepts, and we will perform exactly this evaluation in our future experiments.

swan

Wiki500			BNC		
be	**bird**	C	*have*	**number**	I
be	**black**	P	*have*	**water**	C
have	**chick**	P	*have*	**lake**	C
have	**plumage**	C	*be*	**bird**	C
have	**feather**	C	*be*	**white**	C
restrict	**water**	C	*have*	**neck**	C
be	**mute**	P	*be*	**wild**	P
eat	**grass**	P	*have*	**duck**	I
turn	**elisa**	I	*have*	**song**	I
have	**neck**	C	*have*	**pair**	I

pineapple

Wiki500			BNC		
be	**fruit**	C	*have*	**fruit**	C
be	**sweet**	C	*have*	**leaf**	C
have	**tree**	C	*have*	**plant**	P
have	**export**	I	*have*	**food**	I
divide	**asset**	I	*have*	**end**	I
be	**juice**	C	*have*	**ring**	P
reduce	**may**	I	*have*	**poll**	I
be	**large**	P	*have*	**mint**	I
interfere	**preparation**	I	*have*	**chunk**	P
have	**meaning**	I	*sell*	**shop**	P

screwdriver

Wiki500			BNC		
use	**handle**	C	*have*	**tool**	C
have	**blade**	P	*have*	**end**	P
use	**tool**	C	*have*	**blade**	P
remedy	**problem**	P	*have*	**hand**	I
have	**size**	P	*be*	**sharp**	P
have	**head**	C	*have*	**bit**	P
rotate	**end**	P	*have*	**arm**	I
have	**plastic**	P	*be*	**large**	P
achieve	**goal**	I	*be*	**sonic**	I
have	**hand**	I	*have*	**range**	P

Table 3.8: Top ten returned features and relations for ***swan***, ***pineapple*** and ***screwdriver***.

3.3.4 fMRI activation evaluation

fMRI (functional magnetic resonance imaging) detects and measures human brain activity by monitoring changes in oxygen concentrations of blood in the brain. In an fMRI scan, voxel[14] image data measuring these changes is collected over time from a human volunteer while they perform a given cognitive task or respond to external stimuli. These changes in blood oxygen concentration have long been known to be linked to cognitive processes (Huettel et al., 2009) and fMRI patterns of activation across voxels are widely used to study brain function in response to external stimuli.

Mitchell et al. (2008) adopted the position that the meaning of concrete concepts is encoded in the brain with information associated with basic sensory and motor activities (such as actions involving change to spatial relationships and actions performed on objects) and tested this hypothesis by creating a semantic model trained on a large corpus and used it to predict fMRI patterns of activation from human participants observing several dozen concrete nouns.

fMRI data such as that in the Mitchell et al. (2008) dataset could potentially offer a number of benefits over other evaluation techniques. Unlike property norming data, fMRI data offers direct insight into how the brain is functioning in response to given stimuli, while its multidimensional nature makes it easier to inspect what aspects of meaning a particular model is performing strongly or weakly on, allowing for better control of experimental variation. Assessing our conceptual models' capacity to predict fMRI patterns of activation therefore offers another potential method of assessing the system's output. As far as we are aware, we are the first to evaluate models derived from property extraction in this way.

Semantic models

We consider four different semantic models to compare and evaluate: the Mitchell verb-based semantic model, an SVD model and two versions of our own model (unweighted and weighted). These models were chosen as we were interested in the various kinds of knowledge (part of speech, syntactic and semantic) available in corpora to the extraction process, and the extent to which the use of these types of knowledge affects the quality of the extracted conceptual representations.

The first semantic model we considered was that of Mitchell et al. (2008). This model assumes that sensory-motor information is an important aspect of conceptual representation, and that the information relevant to a target concept's representation

[14]The voxels of an image in three-dimensional space are analagous to the pixels of a two-dimensional image. A voxel is the smallest brain area that fMRI can measure and is typically around 3mm $\times$ 3mm $\times$ 5mm in volume.

Method	Feature Type	POS	Syntax	Semantics
Mitchell et al.	25 verbs	no	no	no
SVD	tuples (content-words)	yes	no	no
Unweighted	triples (properties)	yes	yes	no
Weighted	triples (properties)	yes	yes	yes

Table 3.9: Comparison of the information available to each semantic model.

can be estimated from the concept word's frequency of co-occurrence with 25 sensory-motor verbs (*eat*, *manipulate*, *push*, etc.) in a very large corpus. Our reimplementation of this method used the co-occurrence statistics provided by Mitchell et al. that were extracted from the Google *n*-gram corpus consisting of 1 trillion words of web text.

We also employed the SVD baseline model, as described above (Section 3.3.1). Finally, we used the top 200 triples from our system ranked by frequency (i.e., unweighted) and the top 200 triples after reweighting with the semantic data.

To ensure that the linear regression model for each method would be fitted using the same number of free parameters during training (thereby maximising the comparability of the different methods), we reduced the dimensionality of the generated feature spaces for the SVD method and our two triple-extraction models using Principal Components Analysis (PCA) (Pearson, 1901). The concept-feature extraction frequency matrices for the three models were submitted to PCA, and the first 25 components (i.e., those components which best characterised the variance of the original features) for each model were selected.

Experiment

We are primarily interested in using the fMRI data to evaluate the quality of the different methods for extracting conceptual representations from corpora (rather than investigating methods for predicting fMRI activation). The quality of the predictions generated for the concepts using each model can be adopted as an index of semantic model performance.

A key difference between the Michell et al. model and our models is that while Mitchell et al. posit that certain sensory-motor function verbs can act as important features of concepts, our models instead place more importance on intrinsic semantic properties for describing concepts.

Table 3.9 gives a summary comparison of the different models in terms of whether or not each uses part of speech data, syntactic information (i.e., GRs), and semantic filtering.

It should be noted that the BNC corpus (used with the SVD model and our triple-

extraction method) is 10,000 times smaller than the corpus from which the Mitchell et al. feature vectors are derived. As such, the semantic representations we extract with our method need to make better use of the data available in the corpus if they are to compete with the verb-based features used by Mitchell et al.'s method.

Results

The accuracy for each of the four methods was evaluated using a leave-two-out validation paradigm. There are 1,770 possible pairs of concepts that can be drawn from the set of 60 concept stimuli. Training was performed separately for each participant and for each of the 1,770 held-out pairs. Given a particular participant and held-out pair, for each voxel v, we fit the activation at that voxel to the set of 58 training items with multiple linear regression, using as predictor variables the elements of the 25-dimensional feature vectors associated with each of the 58 concepts. Training therefore yields a set of 25 β-coefficients, which can be used to generate a prediction for the activation y_v of voxel v for the held-out word w using the equation

$$y_v^{\text{pred}} = \sum_{i=1}^{25} \beta_{v,i} f_{i,w} \tag{3.2}$$

where $f_{i,w}$ is the i^{th} element of the feature vector for word w (see Mitchell et al. (2008) for details). Over all voxels, this method gives a prediction for the activation with respect to the held-out word w which can then be compared to the observed activation for that stimulus.

Rather than comparing the activity between predicted and observed images using all voxels, we compared images using only the 500 most stable voxels. For each participant, the 500 most stable voxels were the voxels which gave the most consistent pattern of activation across the six presentations of all 60 stimuli.

We calculated similarity between predicted and observed images using both cosine and Pearson correlation; we report the results using Pearson correlation here, as this measure consistently gave slightly better accuracies for each of the four models (the results were very similar using the cosine measure). Following Mitchell et al. (2008; supplementary material), a match score for each held out pair w_1 and w_2 was calculated as the sum of the similarities between the correctly aligned predicted and observed images:

$$a = \text{sim}(w_1^{\text{pred}}, w_1^{\text{obs}}) + \text{sim}(w_2^{\text{pred}}, w_2^{\text{obs}}) \tag{3.3}$$

Similarly a mismatch score was calculated as

$$b = \text{sim}(w_1^{\text{pred}}, w_2^{\text{obs}}) + \text{sim}(w_2^{\text{pred}}, w_1^{\text{obs}}) \tag{3.4}$$

Method	P1	P2	P3	P4	P5	P6	P7	P8	P9	Mean
Mitchell et al.	0.84	0.83	0.76	0.81	0.79	0.66	0.73	0.64	0.68	0.75
SVD	0.82	0.67	0.79	0.83	0.74	0.64	0.64	0.70	0.75	0.73
Unweighted	0.76	0.64	0.73	0.64	0.68	0.57	0.59	0.58	0.66	0.65
Weighted	0.82	0.72	0.76	0.83	0.73	0.65	0.68	0.51	0.76	0.72

Table 3.10: Accuracy results for the four semantic models in the fMRI evaluation.

Cases where the match score is greater than the mismatch score (i.e., $a > b$) count as successes for the model (i.e., the model correctly identifies the two predicted images). Otherwise there is a failure by the model (i.e. the model identifies the observed image for w_1 as being w_2 and vice-versa).

Table 3.10 presents the results of the leave-two-out cross-validation evaluation, giving the proportion (across all 1,770 pairs) of predicted images for the held-out pairs that were correctly matched to the observed images.[15] The original Mitchell et al. (2008) model has the best mean performance, followed by the SVD model, the weighted triple-extraction method, and finally the unweighted triple extraction method. Across the nine participants, there is no significant difference in the accuracy of the three best-performing methods ($|t(8)| < 1.49$, $p > 0.17$, for all pairwise paired t-tests between Mitchell et al. (2008), SVD, and weighted triple extraction). The unweighted triple extraction method performs significantly worse than the other three methods ($|t(8)| > 3.00$, $p < 0.02$).

That there is no difference between the performance of the Mitchell et al. (2008), SVD and weighted triple extraction methods is surprising, given the different kinds of information that are available to the three models. In particular, the models that automatically acquire very general and semantically unconstrained property-based representations perform as well as the model that uses a set of manually selected sensory-motor verbs, even though the representations generated for these models are derived from 10,000 times less corpus data. This is an interesting finding, given that previous research has suggested that aspects of meaning defined by sensory-motor verbs may have a somewhat distinctive role to play in predicting the fMRI activation associated with conceptual stimuli (Mitchell et al., 2008). The results indicate that general and automatic extraction methods—which extract unconstrained representations and

[15] Our results for the Mitchell et al. (2008) method are similar, though not identical, to those reported in that paper (where the reported mean accuracy across all participants is 0.77, using cosine similarity). One possibility for this discrepancy is that our implementation of the method for selecting the 500 most stable voxels may differ slightly from that used by Mitchell et al. (2008; see supplementary material). In any case, the same set of 500 voxels for each participant were used for generating the results of each model presented here, and so we do not believe this should issue affect comparisons of the different models.

make no assumptions regarding the kinds of properties likely to be predictive of neural activation—can do as well as a model that uses manually selected verbs and is thus designed to be optimal for the task in question.

3.3.5 EEG activation evaluation

We also run a similar experiment to that described above, but using EEG activation data. EEG data measures spontaneous voltage fluctuations on the scalp resulting from the activity of neurons in the brain. The advantage of EEG data over fMRI data is that although the latter has very fine-grained spatial resolution in terms of the brain, it has rather coarse temporal resolution. EEG, on the other hand, has low spatial resolution but millisecond-level temporal resolution—making it potentially possible to analyse real-time linguistic processes. One of the key benefits of this method is that it may allow a fine-grained analysis of performance, for example by revealing the classes of properties (part-of, taxonomic, etc.) which a given model is particularly good at extracting.

Murphy et al. (2009) performed an experiment in which seven participants performed a silent naming task. Participants were each presented with a series of 60 greyscale photographs of items from two classes: tools (e.g., ***spanner*** and ***scissors***) and land mammals (e.g., ***squirrel*** and ***camel***) and asked to think of the name of the object represented in the stimulus image. Thirty stimulus images from each of the two classes were presented, and each image was presented six times in total (the images were presented in a random order), giving a total of $30 \times 2 \times 6 = 360$ presentations per participant in total. According to a post-session questionnaire, participants agreed on image labels in approximately 90% of cases. For full technical details, see Murphy et al. (2009). It is this EEG dataset that we employ to test our system.

EEG data is relatively noisy, therefore we needed to select those features likely to give a consistent correspondence with brain activity encouraged by stimulus words. Murphy et al. (2009) used a combination of three strategies (correlation, noisiness and distinctiveness) to select their features. We derived an empirically favourable weighting of these three strategies, and also derived an empirically optimal number of stimulus signals to employ as follows: we used a support vector machine to perform the mammal/tool category classification task, varying the size of the activation feature-set from 10 to 1000. Each set was ranked across the three criteria using ten-fold cross validation, and a weighted linear combination of these orderings was then employed to obtain an overall ranking—we picked those combinations which minimised the error rate across presentations. The best SVM for the classification task was achieved when taking a weighted average ranking of the three strategies as defined by Murphy et al.

to rank a feature, x:

$$\text{rank}(x) = (2\text{corr}(x) + \text{nois}(x) + \text{distinc}(x))/4 \tag{3.5}$$

Here, $\text{corr}(x)$ is the stability of x across presentations as defined by a correlation measure, $\text{nois}(x)$ is "the amount of power variation seen across presentations of the same stimulus" and $\text{distinc}(x)$ is "the amount of variation in power estimates across different stimuli." Only the top 25 features for each participant are used.

Having derived an optimal feature-set as described above, we tested four corpus-extracted models of semantic representation on their ability to predict observed EEG activation patterns.

As described in the previous section, Mitchell et al. (2008) employed 25 manually selected verbs as their corpus features. Murphy et al. used the same principle to choose 25 Italian verbs (see Murphy et al. (2009) for full details), and employed the Yahoo API to generate co-occurrence statistics for the 60 target concepts. Each concept was represented by a vector recording the number of times it co-occurred in the Yahoo count within a span of 5 words left and right of each of their chosen verbs. We refer to this first model as 'Yahoo Mitchell'.

Murphy et al. (2009) employed the SSLMIT Repubblica corpus (Baroni et al., 2004) containing 400 million tokens of newspaper text to create various word-space models. Their best word-space model adopted a window-based approach, in which each target noun was represented by its co-occurrence with every verb within the same sentence and where no more than one other noun lay between the target and the verb. This large feature matrix was reduced using singular value decomposition, and the top 25 left singular vectors were chosen, weighted by their corresponding singular values to give their model. We call this second model 'Repubblica Murphy'. We used the SVD baseline model as the third model, and the fourth and final model employed our best extraction method, considering features and relations and using hierarchical clustering, as described in Section 3.2.

The Yahoo Mitchell model has full coverage of all 60 concepts in Italian, while the Repubblica Murphy method has coverage for 57 of the concepts.[16] This is due to their extraction method missing some multi-word units. Murphy et al. supply English translations for all 60 of their concept words which we employed when training the models. Our method has coverage for 53 of the 60 translated concepts, due to the omission of certain compound nouns and data sparsity (our corpus, the BNC, is 4 times smaller than the Repubblica corpus). We therefore use a combination of synonyms and sub-

[16]The original paper reports coverage for 58 concepts however the listing we obtained indicated coverage of only 57 concepts.

sumptions to generate properties for the missing words. For ease of comparison, we evaluate all models on the 57 concepts which the Repubblica Murphy model was able to generate feature sets for.

We again follow the leave-two-out paradigm of Mitchell et al. (described above), but apply it to the EEG power data features (with varying spatial, temporal and frequency values as opposed to the fMRI data of specific voxels) to evaluate the quality of the semantic models. The preliminary results indicate that our own method is performing at least comparably to Murphy et al.'s models. Across the seven participants, there is no significant difference in the accuracy of the four methods ($|t(6)| < 0.63$, $p > 0.62$, for all pairwise paired t-tests between Mitchell et al., SVD, Repubblica Murphy and our weighted triple extraction). The results are shown in Table 3.11. It may be the case that using a support vector machine to ascertain the optimal weighting of the three strategies as well as the number of features used is not the best approach. Indeed, it seems rather surprising that a mere 25 features from the EEG activation data would be able to capture the full extent of cognitive activity relevant to the semantic processes which we are seeking.

Method	P1′	P2′	P3′	P4′	P5′	P6′	P7′	Mean
Yahoo Mitchell	0.622	0.424	0.462	0.504	0.576	0.535	0.542	0.523
Repubblica Murphy	0.519	0.476	0.519	0.477	0.428	0.504	0.612	0.505
SVD Baseline	0.443	0.473	0.425	0.466	0.434	0.555	0.560	0.480
Our method (weighted)	0.569	0.520	0.593	0.460	0.423	0.525	0.539	0.518

Table 3.11: Accuracy results for the four semantic models in the EEG evaluation.

3.3.6 Correlational statistics evaluation

It is widely agreed that there are structural differences in the way living and non-living things are represented in the brain (Caramazza and Shelton, 1998; McRae et al., 1997). According to these theories, inter-correlation of properties of living things should be stronger than that between properties of non-living things and living things also tend to have more shared properties. For example, *has* **a tail** is a property which is common to a large number of animals, and we therefore might hope that our model will learn that animals which possess the property *has* **a tail** also often have such properties as *has* **claws** and *has* **fur**. The same is not true for non-living things—it is much harder to generalise properties such as *made of* **plastic** to extrapolate other predictions from them since they are often not very correlated with other properties. We would therefore like our corpus-extraction model to emulate these correlational characteristics. One set of measures designed to measure these correlations, known as the Conceptual Structure

Account (CSA) variables (Randall et al., 2004; Taylor et al., 2008), have already been calculated on the McRae norms. The variables extracted were:

- **NOP**, **NODP**, **NOSP**. The number of properties, number of distinctive properties and number of shared properties, respectively, in a concept's representation. A property is defined as 'distinctive' if it is shared by one or two concepts, and 'shared' otherwise.
- **Proportion shared**. The proportion of a concept's properties which are shared.
- **Mean distinctiveness**. The mean average distinctiveness of a concept's properties, where the distinctiveness of a property is defined as the reciprocal of the number of concepts which share that property.
- **Mean correlational strength**. The mean Pearson correlation between significantly correlated pairs of shared properties of the concept.

Measure	**Correl.**	p
Number of properties	0.203	< 0.001
Number of distinctive properties	0.168	< 0.001
Number of shared properties	0.113	$= 0.012$
Mean distinctiveness	0.155	< 0.001
Proportion of shared properties	0.166	< 0.001
Mean correlational strength	-0.118	$= 0.009$

Table 3.12: Evaluation in terms of the CSA variables: correlations.

We calculated conceptual structure variables from our own extracted properties using a method equivalent to that used by Devereux et al. (2009),[17] then correlated the measures with those of the McRae norms.

To perform this analysis, we used the properties from our best-performing method (in terms of precision) for the Wiki500 corpus. We used production frequency vectors for each concept, normalised to unit length, to calculate the conceptual structure measures on both the anglicised McRae norms and the output triples, excluding a small number of concept words with multiple meanings (e.g., ***bat***, ***fan***). In Table 3.12 we can see the correlations between the extracted properties and the McRae norms for the various CSA variables. In Table 3.13 we compare both the mean values and differences (t-test) between living and non-living groups of concepts across both sets of properties.

Our results show significant correlation for five of the six conceptual structure variables, which would indicate that the extracted properties are capturing at least some

[17]Thanks to Barry Devereux for performing these calculations.

Measure	M_{L}	M_{NL}	t	p
McRae norms				
Number of properties	13.0	12.0	3.06	$= 0.002$
Number of distinctive properties	2.78	4.64	-8.58	< 0.001
Number of shared properties	10.2	7.39	10.79	< 0.001
Mean distinctiveness	0.23	0.39	13.26	< 0.001
Proportion of shared properties	0.79	0.62	11.93	< 0.001
Mean correlational strength	0.28	0.26	3.12	$= 0.002$
Extracted triples				
Number of properties	242.99	260.87	-0.91	$= 0.366$
Number of distinctive properties	122.03	133.55	-1.05	$= 0.295$
Number of shared properties	120.96	127.33	-0.71	$= 0.475$
Mean distinctiveness	0.48	0.48	-0.42	$= 0.667$
Proportion of shared properties	0.52	0.51	1.19	$= 0.234$
Mean correlational strength	0.24	0.28	-5.56	< 0.001

Table 3.13: Evaluation in terms of the CSA variables: living (M_{L}) and non-living (M_{NL}) differences.

aspects of the conceptual structure found in the McRae property norms. However, some of these correlations are weak, and we do not observe the differences between living and non-living concept groups seen in the McRae norms. What we are hoping to demonstrate through this evaluation is the potential utility of using conceptual structure variables to evaluate the semantic models; it is our expectation that improvements in our extraction method would propagate through to improvements in the conceptual structure statistics. Indeed, if we can show that the conceptual structure characteristics of the extracted properties are similar to those of the McRae norms, then this would suggest that the properties are able to capture important structural properties of the conceptual space. This space acts as a layer of separation away from the McRae norms since (as we have already discussed) comparing directly with them is problematic. Even if the features and relations themselves are not identical to the McRae norms on the surface, the underlying structure may in fact be similar. However, given that this evaluation abstracts out our system's output by collecting and grouping the extracted properties, it will be difficult to use it to guide future system development. We will therefore continue our work employing more direct evaluation techniques.

3.4 Discussion

This chapter exhibits a practicable initial system for the extraction of property norm-like information from large bodies of text. We explored a number of different areas of interest, for example, how such information might appear implicitly and explicitly in

text, and the syntactic relationships which are likely to flag such information. We also demonstrated how corpus-choice (be it encyclopedic or general) can affect the system's output. We have shown that using semantic class information to reweight the triples can further boost performance; as one might perhaps expect, due to the hierarchical nature of WordNet, hierarchical clustering works best for this (compared to NMF and k-medoids clustering). Our best F-scores of 0.2593 and 0.1471 (evaluating with and without relation terms, respectively) offer targets for future extraction systems and we will use these scores as a baseline in future experiments.

In this chapter, we have also introduced a number of novel evaluation techniques. The gold standard evaluation using the ESSLLI set will become our reference evaluation for the experiments that follow, however it does suffer from a number of deficiencies. In light of this, we briefly and qualitatively examined our output to investigate the potential benefits of performing direct human evaluations on our system; this analysis indicated that this would indeed be a viable evaluation method, not least because of the significant discrepancies between the gold standard and our own classification. We also introduced two evaluations relating to brain activity. These unfortunately proved inconclusive in terms of differentiating the output of our system from other potential models of the brain—we found no significant differences between the various considered models' performance. These evaluation techniques ought not to be discounted completely, but it is our suspicion that both the models themselves and the methods of brain activity measurement might benefit from further refinement before they can be used as a true evaluation of conceptual property extraction performance. Finally, we evaluated our system's output with a number of correlational statistics; again these results proved promising in that significant correlations, although weak, were present, however we did not see expected differences between living and non-living categories of concept.

Chapter 4

Automatic extraction system

In this chapter, we propose an improved version of our pilot system. Our method extracts candidate triples from two corpora, parsed this time using the C&C parser, with a new set of syntactically and grammatically motivated rules, then reweights triples with a linear combination of four statistical metrics. We illustrate the value of these metrics after our candidate feature extraction stage, and demonstrate their ability to upweight more human-like features. In addition to lexical comparison with norms derived from human-generated property norm data (as in our previous chapter) we also assess our system output in three new ways: direct evaluation by four human judges, semantic distance comparisons with WordNet similarity data and human-judged concept similarity ratings. Our system offers a viable method of plausible triple extraction: a lexical comparison shows comparable performance to the current state of the art whilst subsequent evaluations exhibit the human-like character of the generated properties. This chapter contains work from our journal paper, *Automatic extraction of property norm-like data from large text corpora* (Kelly et al., to appear).

4.1 Data

4.1.1 Recoded norms

As in our previous experiment, we use a recoded version of the British English McRae norms for system training (see Section 3.1.1).

4.1.2 Corpora

We use the same three corpora (Wiki500, Wiki100K and BNC) as in our previous experiment.

4.1.3 Parser

For these experiments, we employed the C&C POS tagger and parser (Curran and Clark, 2003; Clark and Curran, 2007a) to extract both grammatical relations (GRs) and part of speech (POS) tags from the sentences within the corpora. As in RASP output, GRs denote the functional relationships between different words within a sentence, offering a structured representation of the underlying grammatical organisation of a given sentence. The C&C dependency parse output contains, for a given sentence, a set of GRs forming an acyclic graph whose nodes correspond to words from the sentence, with each node also labelled with the POS of that word. Thus the GR-POS graph interrelates all lexical, POS and GR information for the entire sentence. It is therefore possible to construct a GR-POS graph rooted at the target term (the concept in question), with POS-labelled words as nodes, and edges labelled with GRs linking the nodes to one another.

As already mentioned, the RASP parser (Briscoe, 2006) has been used in previous work (our pilot system and that of Devereux et al. (2009)). We have switched to C&C because it has been shown to have better parser accuracy over RASP overall (Clark and Curran, 2007b). Specifically, it has also been shown to outperform RASP on a majority of the grammatical relation types that we will employ in our rules (e.g., direct and indirect objects, non-clausal modifiers and subjects). It is also a lexicalised-grammar parser: it takes into account surface-level lexical information when predicting part of speech tags and grammatical dependencies (something RASP ignores) and it parses text much more quickly than RASP. We parsed all three corpora using C&C.

4.2 Method

Our system works in two main stages. In the first, the C&C parse generates a list of (usually binary) grammatical dependencies between the constituent words. From this list we may construct a GR graph representing the grammatical structure of that sentence. Using a series of rules, we select those paths through the graph containing relations and features which are likely to approximate property-based conceptual representations. Our rules take into account such information as the nature of the GRs in the path, the part of speech tags of the concept, relation and feature as well as path-length information. We place an emphasis on ensuring the relations/features we extract are linguistically motivated. In the second stage, the system weights and ranks the extracted triples by way of a linear combination of four statistical metrics, selected to maximise the possibility that higher-ranked properties will emulate human-generated norms. For example, we choose to downweight properties shared across a very large

number of concepts, as these extremely common (and therefore highly general) properties are unlikely to be cited for any concepts (e.g., *be* **used**, *do* **have**).

4.2.1 Extraction method

Our system is outlined in Figure 4.1. The input to our system consists of a) the set of concepts for which we aim to find properties and b) C&C-parsed sentences from the chosen corpus. The output is a ranked list of concept-relation-feature triples.

Corpus processing

Our system executes two passes over the corpus. The first pass is designed to extract a list of strongly associated words as potential features for each concept to be used as input into one of our more noisy rules (Rule 8) in the second stage. This is done by examining only extremely short grammatical relation paths through the GR-POS graph (i.e., finding modifying nouns and adjectives, indirect objects and possessives relating to the concept). For example, sentences including phrases such as *it attacked the penguins' eggs* and *a penguin egg was found* would indicate, through the possessive and noun-noun compound constructions, that **egg** is a potential feature of ***penguin***.

The second pass employs a manually compiled rule-set (which we describe in the next section) to conduct a breadth-first search over all directed paths rooted at the target concept, logging each time a rule is fired. This process generates candidate concept-relation-feature triples, as well as their frequency of instantiation (according to our rules) across the corpus. One of the rules (Rule 8) will only fire if the found feature appears in the potential feature list, generated in the first pass.

Extraction rules

Our rules from the second pass were constructed in a similar way to those in our pilot system, namely by taking a sample of concepts and their corresponding features from the McRae norms and then examining sentences from the Wiki500 corpus containing a concept and one of its features. Those sentences containing an instantiation of a likely triple would have the path linking the concept and feature through their GR-POS graph examined for a pattern of GRs and POS tags which would be strongly suggestive of a true relation/feature. Provided this pattern was not subsumed by any pre-existing rules, a new rule would be generated from it. We note that our rules do not explicitly take negations along the path into account. We include an outline of all of our rules in Table 4.1. For an in-depth explanation of the various POS and GR tags referenced in the rules, see Jurafsky and Martin (2000) (Appendix C) and Briscoe (2006) respectively.

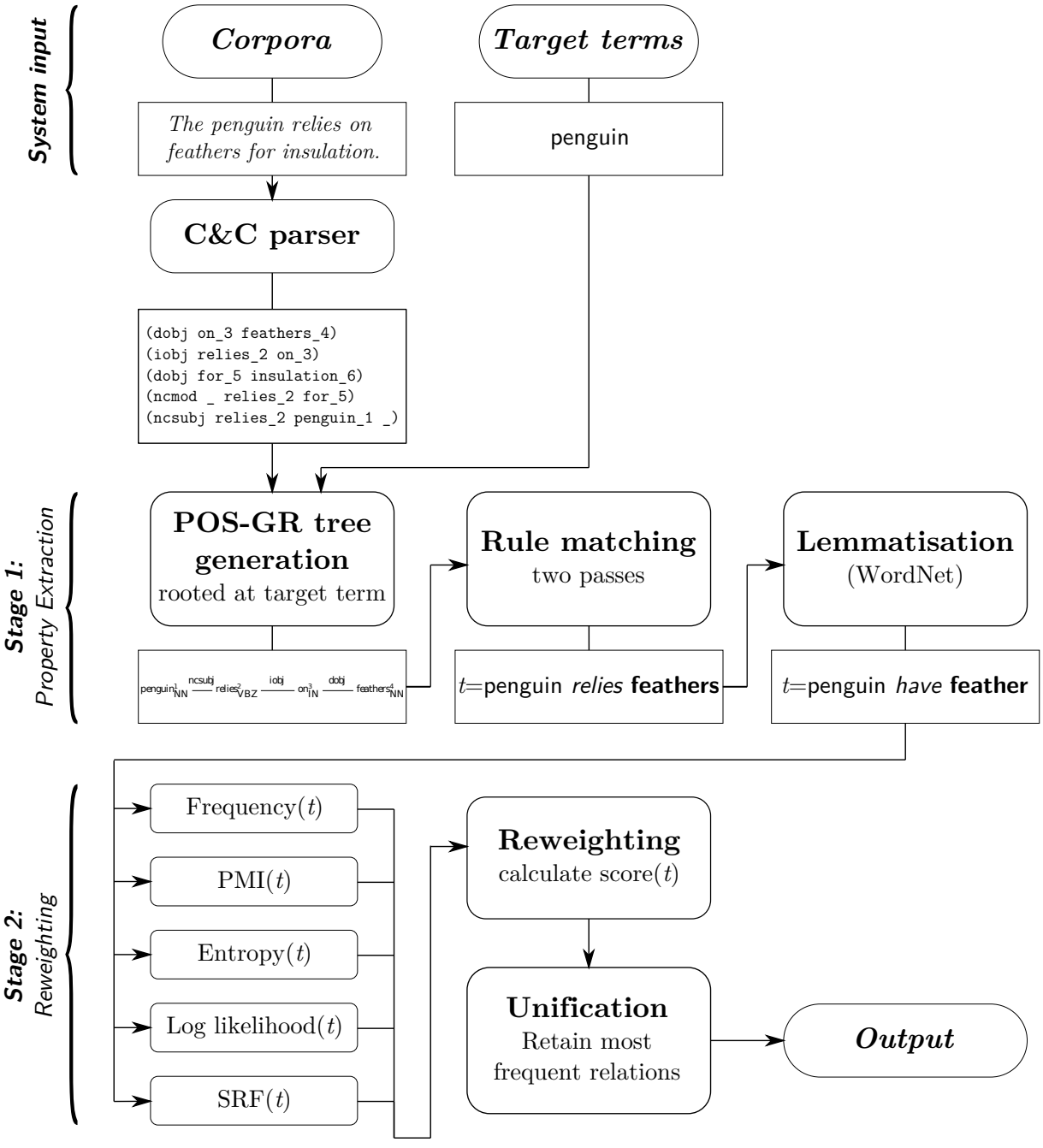

Figure 4.1: An overview of our automatic extraction system, outlining the system input and two main stages: 'Property Extraction' and 'Reweighting'.

Table 4.1: Our 12 rules with the rule's maximum path length (M), frequency information of rule-firing on the Wiki500 corpus, a description of the rule itself, an example sentence and the resulting output triple found from applying the rule to the sentence.

ID	M	Freq.	Rule	Sentence	Triple
1	1	1,060	If feature has a VBG ("being") tag and is linked to concept by a ncmod (non-clausal modifier), xmod (predicative relation modifier), cmod (clausal modifier), or pmod (prepositional modifier) GR then relation is *do*.	American bison grazing in Custer State Park in South Dakota.	***bison*** *do* **graze**
2	1	7,848	If feature is a verb and linked to concept by a ncsubj R (non-clausal subject relation) GR then relation is *do* (unless tag is VBG ("being"), then relation is *be*).	A chain is usually made of metal.	***chain*** *be* **metal**
3	1	16,232	If feature has a NN (common noun) or JJ (general adjective) tag and is linked to concept by a ncmod N (non-clausal modifier) GR then relation is *be*.	In mechanical clocks this is either a pendulum or a balance wheel.	***clock*** *be* **mechanical**
4	1	6,953	If feature has a NN (common noun) tag and is linked to concept by a ncmod R (non-clausal modifier) GR then relation is *have*.	Coconut water can be used as an intravenous fluid.	***coconut*** *have* **water**
5	3	4,445	If feature has a JJ (general adjective) tag and is linked to the rest of its graph by a xcomp (unsaturated VP complement relation) or ncmod (non-clausal modifier) relation then relation is *be*.	Chains can also be decorative as jewellery.	***chain*** *be* **decorative**
6	3	3,650	If feature has a VBN (past participle verb) tag then relation is *be*.	Carrot flowers are pollinated primarily by bees.	***flowers*** *be* **pollinated**
7	3	242	If feature has a NN (common noun) tag then relation is *be*.	The cathedral often being a large building serves as a meeting place.	***cathedral*** *be* **building**

Table 4.1: (continued)

ID	M	Freq.	Rule	Sentence	Triple
8	4	7,572	If feature has a NN (common noun) tag and is linked to its graph by a ncmod (non-clausal modifier) relation then relation is only verb in path to feature.	A chain may consist of two or more links.	***chain*** *consist* **links**
9	4	2,166	If feature has a NN (common noun) tag and is linked to graph by a xcomp (unsaturated VP complement relation) tag then relation is closest verb in path to feature.	Airport trains are trains within airports that transport people between terminals.	***trains*** *transport* **people**
10	4	6,181	Feature has a NN (common noun) tag and its relation has a VVN (past participle) tag and the feature is linked to the relation by a xcomp (unsaturated VP complement relation) GR.	The tiny pharaoh ant is a major pest in hospitals and office blocks.	***ant*** *be* **pest**
11	∞	11,449	Feature has a NN (common noun) tag and final GR is dobj (direct object) and penultimate GR in path is iobj (indirect object) then relation is the penultimate node.	Alligators are native to only two countries: the USA and China.	***alligators*** *native* **usa**
12	4	1,128	If feature has a JJ (general adjective) tag and relation node is a verb then relation is that verb.	Tigers for the most part are solitary animals.	***tigers*** *be* **solitary**

In addition to extracting ***concept*** *relation* **feature** triples, when creating our rules we also place an emphasis on extracting behaviour properties, which appear throughout the McRae norms (e.g., ***penguin*** *beh* **waddles**). This is similar to the model of Baroni et al. (2009), although our rules aim to extract behaviours explicitly exhibited by the concept at hand rather than actions merely associated with the concept. In other words, we are aiming to de-emphasise behaviors such as ***motorcycle*** *beh* **ride** and ***motorcycle*** *beh* **park** to focus on ***motorcycle*** *beh* **travel**, ***motorcycle*** *beh* **cruise** and ***motorcycle*** *beh* **speed**; we would prefer the former relationships to be yielded as ***motorcycle*** *be* **ridden** and ***motorcycle*** *be* **parked**. This involved specifically creating rules (e.g., Rules 1 and 2) which focused on the verbs in the sentence to extract *do* relations (corresponding to the behavioural relations in the McRae norms). Creating these rules was challenging: there are no noun or adjective features to anchor the verb, and it is therefore more difficult to rigorously ascertain which verbs are feature verbs, and which are just incidental given the context. Corpus-based distributional models do not incorporate such distinctions—indeed Baroni et al.'s method fails to distinguish between the behavioural usage of **park** as something a car performs and the associated entity **park**, a place in which a car is parked, conflating the two into the same type-sketch.

Following from this, when constructing our rules it was also important to take account of directionality in the GR-POS graph. This directionality functions as a proxy for the order in which the terms appear in the various grammatical relation slots from the C&C output. Doing this allows us to distinguish between sentences such as "The dog bites the man." and "The man bites the dog." which share an identical grammatical relation structure. This is an essential step for understanding the meaning of a given sentence, in addition to better dealing with passive verb formations. We hope that doing this will also serve to reduce the amount of noise produced in the system's output. Previous systems (e.g., our pilot system and that of Devereux et al.) have not taken this directionality into account.

In constructing the rules in this way, our overriding aim was to ensure that the features and relations extracted were of a high quality, and likely to be true.

To illustrate the mechanics of the system we now give a worked example of how a triple such as ***penguin*** *have* **feather** could be extracted using our system. The following sentence is found in one of the corpora:

The penguin relies on feathers for insulation.

The C&C parse for this sentence yields, along with POS tags for each word in the sentence, the following GR output:

```
(det penguin_1 The_0)
```

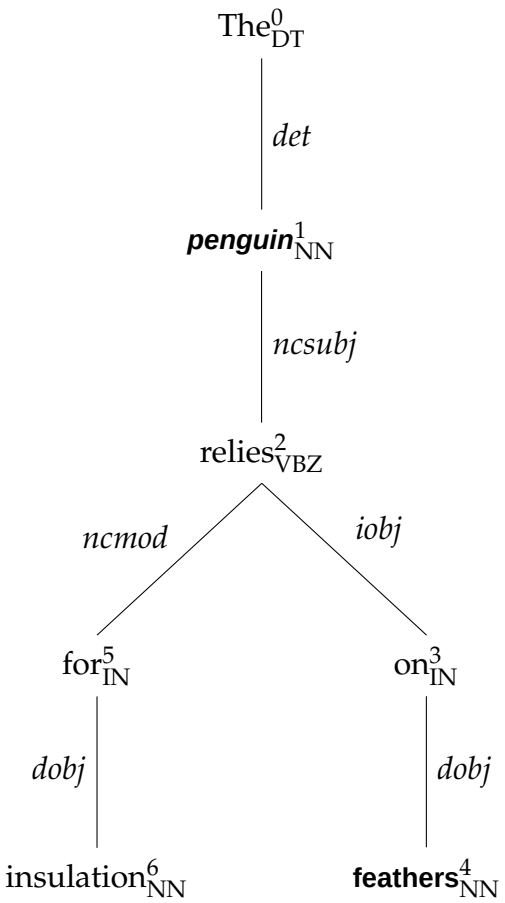

Figure 4.2: A C&C-derived GR-POS graph for the sentence *The penguin relies on feathers for insulation*.

```
(dobj on_3 feathers_4)
(iobj relies_2 on_3)
(dobj for_5 insulation_6)
(ncmod _ relies_2 for_5)
(ncsubj relies_2 penguin_1 _)
```

From this output, we may construct the grammatical relation graph with POS tags as shown in Figure 4.2. One of the McRae concepts is ***penguin***, so we may examine all paths through the graph rooted at the concept. In this example, we find that the path found in Figure 4.3 activates one of our rules (Rule 11), yielding the triple ***penguin*** *relies* **feathers**.

Lemmatisation

We employed the NLTK WordNet lemmatiser (Bird, 2006) to lemmatise all extracted features and relations, using part of speech information to group together various in-

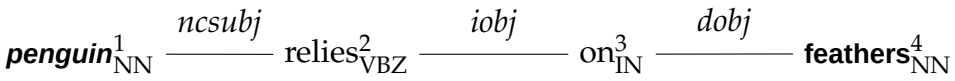

Figure 4.3: A path through the GR-POS graph, activating Rule 11 to derive the triple ***penguin*** *rely* **feather**.

flected forms. This allows us to manipulate semantically identical (or near-identical) words as a single term. The feature is lemmatised as an adjective or a noun unless it is a behaviour feature, whilst the relation head and behaviour features are lemmatised as verbs. For example, if we were to extract the triples ***bull*** *were* **cows**, ***balloon*** – **floats** and ***cake*** *are* **best**, they would be lemmatised to the forms ***bull*** *be* **cow**, ***banner*** *do* **float** and ***cake*** *be* **good** respectively. This stage is important for the evaluation, as it reduces the possibility of triples being marked as incorrect due to inflectional differences. The example triple derived in the previous section (***penguin*** *relies* **feathers**) would thus be converted to ***penguin*** *rely* **feather**.

4.2.2 Reweighting metrics

Our ultimate aim is to go beyond human-elicited norms and extract a full picture of each concept through its properties. That said, we would still want to avoid (or at least downweight) very 'general' properties, i.e., those which are arguably common to all concepts (e.g., ***penguin*** *do* **exist**, ***car*** *be* **thing**). It is relevant and specific properties which we are interested in. Therefore in this stage, we estimate the strength of association between the concepts and features extracted, hypothesising that a higher degree of association will correlate well with human-like norms. We also make use of information directly acquired from the property extraction stage to guide the reweighting, forging a link between the candidate property extraction stage and the reweighting stage. Finally, we again employ our semantic reweighting factor as a parameter in reweighting.

Despite our efforts to ensure that the candidate triples we extract are plausible from a syntactic perspective, it is inevitable that some of them will be incorrect—this is because even grammatically identical constructions often have distinct meanings in different semantic contexts. Consider, for example, the sentences:

Every cloud has a silver lining and *Every beehive has a queen bee.*

Both exhibit identical grammatical structure as demonstrated by their respective C&C parses. However the first is an idiomatic phrase—no cloud possesses a literal silver lining—whereas the second shows a true property of beehives. Context and semantics often greatly affect the meaning of sentences within the corpora. We therefore expect

the output of the first stage of our system to be quite noisy (i.e., producing incorrect triples). We need to ensure that the triples we select from the set of candidates are indeed likely to indicate correct semantic features and relations.

Therefore, the second stage of the system involves reweighting and choosing from the output of relations/features from the first stage in a way that brings to the fore those which we might expect to find in property norm data. In total, we employ four measures to achieve this, and empirically test which linear combination of these four metrics yields the best results. Taking the linear combination in this way also allows us to assess the relative contribution of the metrics towards improving accuracy, illustrating the degree to which each of them is useful.

Pointwise mutual information

Pointwise mutual information (PMI) was first proposed by Church and Hanks (1990) as an objective measure for estimating word association norms. In information theory and statistics it is employed as a metric for measuring the strength of association between two events. It has been widely used in NLP as a measure of word similarity/semantic relatedness (Turney, 2001; Pantel and Lin, 2002). For our purposes, we will employ it as a measure of the strength of association between an extracted concept and its feature.

For a given triple $t = (c, r, f)$, we calculate PMI as:

$$\text{PMI}(t) = \log \frac{\text{freq}(c, f) \times N}{\text{freq}(c) \times \text{freq}(f)} \text{ where } N = \sum_{i \in C} \sum_{j \in F} \text{freq}(i, j) \tag{4.1}$$

Here, C is the set of all extracted concepts and F is the set of all extracted features. In boosting those concept-feature pairs with high mutual information, we hope that more relevant and informative concept-relation-feature triples will come to the fore.

Entropy

We also calculate a novel entropy statistic for each extracted relation/feature pair based on its firing of rules during the second pass of the relation/feature extraction stage. If we define R_t as the set of rules which fire to produce a specific triple t, then we may define the entropy of t as follows:

$$\text{Entropy}(t) = -\sum_{r \in R_t} p(r|t) \log p(r|t) \tag{4.2}$$

where $p(r|t)$ is a probability mass function for the triple t across our rules. We calculate $p(r|t)$ as:

$$p(r|t) = \frac{p(r,t)}{p(r)} = \frac{\text{freq}(r,t)}{\text{freq}(t)} \tag{4.3}$$

where $\text{freq}(r,t)$ is the number of times rule r fires to produce triple t and $\text{freq}(t)$ is the total frequency of triple t. In this way $p(r|t)$ exhibits the usual properties of a probability mass function over the set of rules R_t.

We illustrate this with an example. Suppose we extract four triples (A–D) using four rules (r_1–r_4):

- A: Dog has tail (generated from r_1 only, frequency 2)
- B: Dog is animal (generated from r_1, r_2 and r_3, with frequencies 3, 2 and 5, resp.)
- C: Dog has bone (generated from r_1 and r_3 with frequencies 3 and 2 resp.)
- D: Dog chases cat (generated from r_4 only, frequency 3)

We therefore know:

$$\begin{array}{cccc}
\text{freq}(r_1,A)=2 & \text{freq}(r_2,A)=0 & \text{freq}(r_3,A)=0 & \text{freq}(r_4,A)=0 \\
\text{freq}(r_1,B)=3 & \text{freq}(r_2,B)=2 & \text{freq}(r_3,B)=5 & \text{freq}(r_4,B)=0 \\
\text{freq}(r_1,C)=3 & \text{freq}(r_2,C)=0 & \text{freq}(r_3,C)=2 & \text{freq}(r_4,C)=0 \\
\text{freq}(r_1,D)=0 & \text{freq}(r_2,D)=0 & \text{freq}(r_3,D)=0 & \text{freq}(r_4,D)=3
\end{array}$$

And we also know that:

$$\begin{array}{cc}
\text{freq}(A) = 2+0+0+0 = 2 & \text{freq}(B) = 3+2+5+0 = 10 \\
\text{freq}(C) = 3+0+2+0 = 5 & \text{freq}(D) = 0+0+0+3 = 3
\end{array}$$

From these frequency values we calculate $p(r|t)$ using Equation 4.3:

$$\begin{array}{cccc}
p(r_1|A)=2/2=1 & p(r_2|A)=0 & p(r_3|A)=0 & p(r_4|A)=0 \\
p(r_1|B)=3/10=0.3 & p(r_2|B)=2/10=0.2 & p(r_3|B)=5/10=0.5 & p(r_4|B)=0 \\
p(r_1|C)=3/5=0.6 & p(r_2|C)=0 & p(r_3|C)=2/5=0.4 & p(r_4|C)=0 \\
p(r_1|D)=0 & p(r_2|D)=0 & p(r_3|D)=0 & p(r_4|D)=3/3=1
\end{array}$$

And we can next calculate the entropy for each of the triples:

$$\begin{aligned}\text{Entropy}(A) &= -\sum_{r \in R_A} p(r|A) \log p(r|A) \\ &= p(r_1|A) \log p(r_1|A) \\ &= 1 \times \log(1)) \\ &= 0\end{aligned}$$

$$\begin{aligned}\text{Entropy}(B) &= -\sum_{r \in R_B} p(r) \log p(r|B) \\ &= p(r_1|B) \log p(r_1|B) + p(r_2|B) \log p(r_2|B) + p(r_3|B) \log p(r_3|B) \\ &= -(0.3 \times \log(0.3) + 0.2 \times \log(0.2) + 0.5 \times \log(0.5)) \\ &= 0.4471\end{aligned}$$

$$\begin{aligned}\text{Entropy}(C) &= -\sum_{r \in R_C} p(r) \log p(r|C) \\ &= p(r_1|C) \log p(r_1|C) + p(r_3|C) \log p(r_3|C) \\ &= -(0.6 \times \log(0.6) + 0.4 \times \log(0.4)) \\ &= 0.2923\end{aligned}$$

$$\begin{aligned}\text{Entropy}(D) &= -\sum_{r \in R_D} p(r|D) \log p(r|D) \\ &= p(r_4|D) \log p(r_4|D) \\ &= -(1 \times \log(1)) \\ &= 0\end{aligned}$$

The logic behind taking account of and summing the scores over the number of rules fired is that if a relation/feature pair corresponds to multiple rules, then it is less likely to be a 'false positive'; this follows from Baroni et al.'s observation relating pattern type frequency and token frequency (see Section 2.3.5).

In other words, we wish to adjust the number of rules for a given property by the probability that those rules will fire. Rules which fire less frequently are presumably more difficult to satisfy, and, given that they were constructed specifically to extract accurate relations, they are presumably stronger predictors of an accurate triple. However we also wish to prevent our less common rules from having a disproportionate effect on their upweighting ability due to their relatively low frequency of activation. Using the entropy curve allows us to take into account the incidence of rules firing relative to one another, whilst avoiding outlier rules (for a given concept) unduly in-

fluencing the scoring. This is a novel way of implementing Baroni et al.'s insight that triples derived from multiple rules or patterns are more likely to correspond to true properties.

Log-likelihood ratio

First proposed by Dunning (1993) for use in NLP, the log-likelihood ratio is a measure of the strength of statistical association between words in text. It has been used to contrast the relative frequencies of words in a corpus, and expose and highlight lexical phenomena which are particularly distinctive in large bodies of text—for example words which are under- or over-used compared to the norm. We employ the log-likelihood ratio across the set of concept-feature pairs (and their raw frequency data). The aim is to derive those properties which are particularly distinctive for a given concept, and which will therefore likely be properties of that concept alone. Unlike PMI, the log-likelihood ratio has also been shown to work well under sparse data conditions (Dunning, 1993), making it particularly appropriate for our task.

We generate a frequency contingency table relating to each distinct concept-feature pair across the triples by grouping and summing the production frequencies of triples containing differing relations but the same concept and feature. For each concept-feature pair, this contingency table contains observations across all triples of the occurrence and non-occurrence of both the concept and the feature. For any given concept c and feature f, we define k_{11} to be the total frequency of concept c and f co-occurring across all triples, and k_{12} to be the total frequency of triples with concept c but not with feature f. We define k_{21} as the total frequency of triples with f as feature, but without c as concept, and k_{22} as the total frequency of triples with neither c as their concept nor f as their feature. We then define the log-likelihood ratio for a given triple $t = (c, r, f)$ as:

$$\mathrm{LL}(t) = 2\sum_{i,j} k_{ij} \log \frac{n_{ij}}{m_{ij}} \text{ where } n_{ij} = \frac{k_{ij}}{k_{i1} + k_{i2}} \text{ and } m_{ij} = \frac{k_{1j} + k_{2j}}{k_{11} + k_{21} + k_{12} + k_{22}} \tag{4.4}$$

Semantic reweighting factor

We employ the same method as described in Section 3.2.2 to assess the conditional probability of a feature given that it relates to a specific concept, however this time we estimated this probability distribution using the McRae norms but excluding the ESSLLI set of concepts and their properties. We also altered the cluster sizes so that the concepts and feature terms from the non-ESSLLI recoded norms fell into two sets

of 50 and 150 clusters[1] respectively. We used hierarchical clustering only. Similarity between two words was defined as the maximum value of Lin's similarity metric (Lin, 1998) across all binary combinations of the two words' WordNet senses. As before, if f did not appear in the non-ESSLLI set of features, then f was simply assigned to the cluster it was most similar to.

For example, one of the concept clusters contains (amongst others) the concepts ***asparagus***, ***blueberry*** and ***cranberry***; another contains ***ant***, ***bat*** and ***bear***. Similarly, one of the feature clusters contains **apple**, **berry** and **carrot** whilst another contains **accomplishment**, **achievement** and **award**.

Reweighting

To order the output of our system, we score and rerank the output triples, $t = (c, r, f)$ where c, r and f are a given ***concept***, *relation* and **feature** respectively, as follows:

$$\text{score}(t) = \beta_{\text{PMI}} \cdot \text{PMI}(t) + \beta_{\text{Ent}} \cdot \text{Entropy}(t) + \beta_{\text{LL}} \cdot \text{LL}(t) + \beta_{\text{SRF}} \cdot \text{SRF}(t) + \text{normfreq}(t) \tag{4.5}$$

The multiplicands of the four free variables were normalised by finding, for each concept, the highest (`max`) and lowest (`min`) possible values of each metric across all triples $t \in T$, subtracting `max` from every triple and then dividing the result by the difference (`max-min`). In this way the values of each metric lay between 0 and 1, with, for each concept, a maximum value of 1 for at least one triple and a minimum value of 0 for at least one triple. In cases where the difference was 0, all triples for that metric and concept were assigned zero (and consequently that metric, for that concept, would have no impact on the triple ranking).

Doing this allows a crude assessment of the importance of the metrics relative to one another. We also use a normalised frequency measure for each triple, normfreq(t). In this way, the trivial case when $\beta_{\text{PMI}} = \beta_{\text{Ent}} = \beta_{\text{LL}} = \beta_{\text{SRF}} = 0$ yields a ranking equivalent to ordering by frequency of extracted relations/features alone. We will offer empirically derived values for these parameters in the sections which follow.

Relation unification

The final part of the reweighting stage attempts to ascertain the most likely relation for similar triples. This involves concatenating all triples which share a concept and feature by removing the less frequent of these and summing their scores to the most

[1] We chose these values so that the average cluster size would be around 10.

frequent triple. For example, if we were to have extracted the triples ***penguin*** *splash* **water**, ***penguin*** *swim* **water**, ***penguin*** *live* **water** with scores 1, 4 and 5 respectively, then these would all be combined into a single triple, ***penguin*** *live* **water**, with score 10. In the example above, ***penguin*** *rely* **feather** could be grouped under a triple such as ***penguin*** *have* **feather**, as this would likely be more frequent over the corpus.

4.2.3 Training

To avoid overfitting our system to a specific set of concepts, we will train it (and the free variables in Equation 4.5) on a subset of the concepts. Then we will evaluate our system on an unseen set of concepts. We intend to use the 44 ESSLLI concepts (with relation expansion labels) for the final evaluation, and therefore we employ the remaining 489 McRae concepts to train the system. We do not have an expansion set for these triples, and so must train and evaluate using exact matching on those relations and features found in the McRae norms.

We begin by examining how well our system is performing when applied to the training data prior to the reweighting stage (i.e., when all the β values are 0). These results can be found in Table 4.2. It is clear that ranking the triples by frequency alone does not offer particularly strong results, with F-scores only reaching 0.0358 (with relation) and 0.1235 (features only).

Relation	Corpus	Prec.	Recall	F
With	Wiki500	0.0307	0.0455	0.0358
	Wiki100K	0.0290	0.0449	0.0346
	BNC	0.0270	0.0403	0.0318
	Wiki100K-BNC	0.0348	0.0537	0.0415
Without	Wiki500	0.0909	0.1295	0.1033
	Wiki100K	0.0870	0.1332	0.1035
	BNC	0.1060	0.1590	0.1235
	Wiki100K-BNC	0.1141	0.1764	0.1363

Table 4.2: Precision, Recall and F-scores for all extracted top twenty triples (ranked by frequency) when evaluating against the training (non-ESSLLI) norms, both including and excluding the relation.

Combining corpora

A qualitative analysis of the output indicates that there is actually relatively small overlap in the output of the two larger corpora: there is an average overlap of 27.0% between the output of the Wikipedia set and the output of the BNC set, and an even

smaller average overlap of 14.6% when also taking the relation into account. It therefore again seems worthwhile to evaluate the extent to which the triples from both corpora complement one another by measuring the performance when evaluating on a combination of the two (i.e., combining the output triples and scores from both sets and retaining the top twenty scoring triples from this combined set). The precision and recall results for this extraction set, which we call Wiki100K-BNC, can also be found in Table 4.2. The combination of these two larger corpora offers a slight improvement on the best scores (from the BNC corpus alone), indicating that this is indeed a viable approach.

Parameter estimation

We now evaluate the output generated when reweighting the triples using the linear combinations of metrics described above applied to Equation 4.5 and against the training (non-ESSLLI) norms. Ideally, we would want to see a reasonably high level of recall (i.e., at a level comparable to that of our preliminary system) combined with a higher precision figure. This would indicate that our system has become more discerning in terms of the triples it extracts, and hence fulfills one of the aims of the property extraction rule-set.

Relation	Corpus	β_{LL}	β_{PMI}	β_{Ent}	β_{SRF}	Prec.	Recall	F
With	Wiki500	0.04	0.00	0.00	1.00	0.0330	0.0492	0.0386
	Wiki100K	0.04	0.00	0.02	0.98	0.0414	0.0625	0.0490
	BNC	0.00	0.00	0.04	0.90	0.0367	0.0543	0.0430
	Wiki100K-BNC	0.00	0.00	0.05	0.68	0.0502	0.0764	0.0596
Without	Wiki500	0.00	0.05	0.03	1.00	0.0940	0.1346	0.1071
	Wiki100K	0.03	0.00	0.04	1.00	0.1065	0.1622	0.1265
	BNC	0.00	0.02	0.14	0.76	0.1197	0.1787	0.1394
	Wiki100K-BNC	0.03	0.03	0.09	0.78	0.1339	0.2057	0.1596

Table 4.3: Parameter estimation for our automatic extraction system when evaluating against the training (non-ESSLLI) norms, both including and excluding the relation.

We varied the β values in the range [0,1] with an initial increment of 0.05, and then used the best-performing values to search for local F-score maxima around these values with increments of 0.01. We can see that the reweighting is indeed increasing the resulting F-scores slightly, although not by as much as we might have hoped.

In general, we can see the reweighting favours the SRF metric, however the entropy parameter also figures across seven of the eight different optimised systems, indicating that this is indeed a feasible metric. This might be somewhat surprising, especially

considering how few rules we are employing in the extraction stage. The PMI and LL weightings are more variable (and less important) in their contributions to the 'best' systems, PMI notably not helping at all when reweighting triples with their relation included.

4.3 Evaluation

Having trained our system, we will employ a number of evaluation methods to ascertain the quality of the extracted features and relations. We use three types of evaluation: comparison of extracted ***concept*** *relation* **feature** labels with the ESSLLI expansion set, a semantic similarity task measuring the ability of our extracted labels to predict concept-concept similarity (using both human- and WordNet-derived similarity metrics) and finally a comprehensive direct human evaluation of the generated relations/features.

4.3.1 Gold standard evaluation

In these experiments we choose, in contrast to Baroni et al., to optimise and evaluate our system based on the top twenty extracted triples. We do this because it is simply not the case that all concepts possess only ten properties (the majority of the McRae properties have more, and we have already discussed the incompleteness of the McRae norms). Although this will automatically lower our highest possible scores (the average McRae concept has 14.7 properties), we believe that including the top twenty offers a better insight into actual performance (all the more so when later performing the human evaluation). Doing this also offers us a better picture of how the reweighting stage is affecting the results, since we are considering a larger sample of highly ranked triples. We will also examine performance of our best system for the top ten extracted triples to offer like-for-like comparison with the evaluation of Baroni et al.

Since we are taking the top twenty triples, the results are not directly comparable to those of Devereux et al. (2009) either, who evaluated their system on the top 25% returned properties. Throughout this section we compare our results with the best-performing method from our pilot system: both systems have been evaluated identically. Our pilot system has a similar structure to the new system but the key differences are that it uses the RASP parser rather than C&C, it has a different, more permissive candidate triple-extraction rule-set (which makes only one pass over the corpus), and it reweights its extracted triples using the semantic reweighting factor alone. It was also

both optimised for and evaluated on the ESSLLI set, making it a good 'best-possible' score to aim for.

Pre-reweighting results

Having estimated and fixed the parameters based on the training (non-ESSLLI) concepts, we are now able to evaluate our system using those parameters. As already mentioned, we are evaluating on the top twenty returned triples. We note that when comparing with the ESSLLI gold standard, we are actually incorporating an upper bound for precision of 0.500 as ESSLLI contains only ten properties per concept. We do this because we are aware that the gold standard is incomplete, and therefore it is plausible that we are extracting triples which are indeed correct but will be evaluated as wrong when compared with the ESSLLI set.

Although we are aiming to evaluate our system in its entirety (i.e., when considering the post-extraction statistics), it is illustrative to see how well the initial rule-based extraction method is performing on the 44 ESSLLI concepts compared to our pilot system (i.e., compared to the results in Table 3.6). In Table 4.4 we report the results for all of the system output (ignoring the relation terms). Although precision is still low (between 1% and 4% for our new method), this is because there is no filtering on the output and there are thousands of triples being evaluated for each corpus (42,777 triples for Wiki500, 515,228 for Wiki100K and 568,793 for the BNC corpus); this output is extremely voluminous compared to the number of "correct" triples (440), placing an extremely low upper bound on precision. This is also why the Wiki500 corpus tends to perform better—this corpus is much smaller than the other two and therefore produces fewer triples. These results would appear to indicate that in the initial property extraction, we have increased precision by a notable margin without an enormous loss of recall, especially for the Wiki100K and BNC corpora, when comparing to our pilot system. The change is less impressive for the Wiki500 corpus with quite a strong reduction in recall, but this might be explained by the fact that our method is much more restrictive than our pilot system in its candidate property extraction. As such, the reasonably good results from the preliminary system on the Wiki500 corpus are a consequence of this relatively small and task-specific corpus being more likely to contain correct properties and less noise.

In Table 4.4, we also report results when matching on the top twenty features only, ordered by frequency and prior to any reweighting. Here, we can again see an improvement almost entirely across the board; the new extraction system is clearly in the first instance generating more sensible triples than our pilot system. However, there is a slight reduction in F-score for the Wiki500 corpus. We feel this is again due to a more

	Relation	Corpus	Prec.	Recall	F
Pre-reweighting	With	Wiki500	0.0312	0.0614	0.0413
		Wiki100K	0.0318	0.0636	0.0424
		BNC	0.0341	0.0682	0.0455
		Wiki100K-BNC	0.0375	0.0750	0.0500
		Pilot system	0.0242	0.6515	0.0467
	Without	Wiki500	0.0924	0.1818	0.1223
		Wiki100K	0.1000	0.2000	0.1333
		BNC	0.1420	0.2841	0.1894
		Wiki100K-BNC	0.1341	0.2682	0.1788
		Pilot system	0.1159	0.2326	0.1547
Post-reweighting	With	Wiki500	0.0323	0.0636	0.0428
		Wiki100K	0.0432	0.0864	0.0576
		BNC	0.0557	0.1114	0.0742
		Wiki100K-BNC	0.0602	0.1205	0.0803
		Pilot system	0.1102	0.2210	0.1471
	Without	Wiki500	0.1015	0.2000	0.1344
		Wiki100K	0.1227	0.2455	0.1636
		BNC	0.1420	0.2841	0.1894
		Wiki100K-BNC	0.1489	0.2977	0.1985
		Pilot system	0.1943	0.3896	0.2593

Table 4.4: Precision, Recall and F-scores for all extracted triples, pre- and post-reweighting, when evaluating against the ESSLLI norms, both including and excluding the relation.

restrictive rule-set, which has prevented large numbers of less-than-certain properties from being extracted. However, again since the Wiki500 corpus is smaller and more task-specific, these "less-than-certain" properties are in fact more likely to be relevant, which might explain the higher F-score displayed by our preliminary system.

Matching on features only

We now evaluate the results of reweighting the triples using Equation 4.5 and the optimal β values as listed in Table 4.3. Ideally, we are seeking both higher recall and precision figures.

Results for our system can be found in Table 4.4. The most significant improvements from the reweighting appear in the Wiki100K corpus, where the F-score increases from 0.1333 to 0.1636. It is also interesting to note that the reweighting has not affected the results of the BNC output; although the triple output has changed due to the reweighting, the number of correct triples has not.

Matching on features and relations

We are not solely focusing on the features extracted; we also want to evaluate the relations. As already mentioned, matching on paired features and relations is a much more challenging task than matching on features alone, made all the more difficult by the fact that we do not have a synonym-expanded set of relations to evaluate on—we are evaluating directly on lemmatised versions of the relations found in the McRae norms. Results can also be found in Table 4.4.

The system is not performing quite as well as our 'best-possible' method from our pilot system. We believe this is for a number of reasons: for one, the 'best-possible' method did not include a blind training phase. Instead the method was optimised directly against the evaluation set, both by varying cluster size and clustering technique, to yield the 'best-possible' results against the ESSLI gold standard. This new system is also more conservative in the relations it is extracting, and more likely to only extract relations which it is confident in. Furthermore, as the ESSLLI expansion set we are comparing with does not include synonyms for the relation verbs, it is possible that the final step of the first stage (in which we attempt to group similar triples) may be backfiring, in that although it is upweighting correct features (as demonstrated by our earlier results), it may be retaining an 'incorrect' relation, and thus not performing as well as our benchmark system (which did not incorporate such a step). For example, ***helicopter*** *have* **pilot** is the highest rated triple with **pilot** as a feature, and hence subsumes all instances of the correct triple from the ESSLLI set, ***helicopter*** *require* **pilot**. In other words, as this system retains only the highest-scoring relation when grouping triples with differing relations but the same features, exactly emulating the specific McRae-derived relations is challenging. This again demonstrates the pitfalls associated with this particular evaluation technique.

4.3.2 Human-generated semantic similarity comparison

Given the issues associated with calculating precision and recall scores directly from the output, we use an additional, alternative approach to calculate how semantically meaningful the extracted triples really are. To do this, we evaluate the triples' capacity to predict similarity between words, using human similarity judgements.

We asked five native English speakers to rate the similarity of 90 concept pairs, where both concepts in all the pairs were drawn from the ESSLLI set. The 90 pairs correspond to ten concept pairs chosen at random from their banded WordNet similarity score, based on Leacock and Chodorow's normalised path length WordNet similarity measure (Leacock and Chodorow, 1998). In other words, there were ten concepts with

score 0–0.1, ten with score 0.1–0.2 and so on. There were no pairs with similarity of 0.9 or above.

The raters were given instructions explaining the task, included in Appendix A, Section A.1. They were then presented with each concept pair, one by one, and a scale of 1 to 7 and asked to rate how similar the two concepts were (instruction text in Appendix A, Section A.2).

The average Pearson coefficient of correlation across the five judges (considering all pairwise combinations) was 0.82.

Using these scores we constructed a vector of dimensionality 90, V_{Human} containing the averaged human-generated similarity scores between the 90 concept pairs. We normalised each score so that it lay between 0 and 1 (i.e., 'very dissimilar' pairs received a score of 0, 'very similar' pairs received 1 and the remaining scores were distributed evenly across the interval). The Pearson coefficient of correlation of their scores with the WordNet semantic similarity scores was 0.75.

To compare our system with these ratings, we wish to approximate the similarity between the 44 ESSLLI concept words given a set of top twenty output triples for each concept. From this we will be able to extract similarity vectors V corresponding to the 90 pairwise human comparisons. To achieve this, we begin by constructing a vector space of dimension D, where D is the number of distinct properties across the 44×20 triples. Then for each of the 44 concepts, we generate a concept-score vector with twenty non-zero entries by inserting the triple scores, score(t), into their correct entries in the concept-score vector. We may then construct a 44×44 symmetric pairwise similarity matrix across the concepts by calculating the cosine similarity between their concept-score vectors. From this we can extract a similarity vector, V, for the 90 concept pairs.

We calculate eight such matrices (both including and excluding the relation term from each concept's triple, across the four corpora). We similarly generate two such matrices from the McRae norms (one using the full text of the property norms as the concept vectors' dimensions, the other using only the features), using the norm production frequencies (in place of score(t)) as entries in each concept's vector.

As already mentioned, we may then report the correlation between the V_{Human} vector and the similarity vectors V. The results can be found in Table 4.5. The confidence intervals, calculated using Fisher transformations (Fisher, 1915), are given are at the 95% level of confidence, and two-tailed $p < 0.05$ for all the correlation calculations. We first notice that given the various vector dimensionalities (D) we appear to be extracting a larger number of distinct features/relation-feature pairs than the McRae norms; this can in part be accounted for by the fact that the McRae norms contain fewer prop-

erties for each ESSLLI concept (with an average of 16.0 per concept rather than our extracted twenty per concept). In other words, we would expect the D_{McRae} figures to be around 20% smaller than the other dimensionalities. In actual fact the discrepancy is closer to 35% (both when including and excluding the relation). This would indicate that our system is extracting a more diverse set of properties than those that appear in the McRae norms.

Our results show that the matrices derived from the BNC triples appear to be the best predictor of concept-concept similarity (both when including and excluding the relation terms), showing the highest overall correlations with the human evaluations.

Relation	***V***	***D***	***r***	**Conf. Int.**
With	McRae	410	0.7853	[0.691, 0.854]
	Wiki500	712	0.3194	[0.120, 0.494]
	Wiki100K	626	0.3927	[0.202, 0.555]
	BNC	601	0.5625	[0.402, 0.689]
	Wiki100K-BNC	586	0.5452	[0.381, 0.676]
Without	McRae	355	0.7874	[0.693, 0.855]
	Wiki500	626	0.4684	[0.289, 0.616]
	Wiki100K	533	0.4897	[0.314, 0.633]
	BNC	542	0.6655	[0.532, 0.767]
	Wiki100K-BNC	524	0.6305	[0.487, 0.741]

Table 4.5: Pearson correlation results between the V_{Human} vector and the similarity vectors V (and their vector dimensionalities D) from our best automatic extraction systems as reported in Table 4.4.

4.3.3 WordNet semantic similarity comparison

We next compare the output with the semantic similarity predicted by WordNet across all pairwise combinations of the 44 concepts. To achieve this, we construct a baseline matrix, M_{LC}, containing similarity scores between all binary combinations, again using the Leacock and Chodorow WordNet similarity measure. We normalised the returned values by dividing through by the maximum possible value, ensuring that all values lay in the range [0,1].

The Frobenious norm of a matrix X is defined:

$$||X||_{\mathbf{F}} = \sqrt{\sum_{i,j} |x_{ij}|^2} \tag{4.6}$$

We can calculate the Frobenious distance between two matrices X and Y as $||X - Y||_{\mathbf{F}}$. A lower Frobenious distance between two similarity matrices implies that they are closer to one another, as it is the matrix equivalent of calculating the Euclidean distance between two points. Results measuring the Frobenious distances between the various matrices and M_{LC} are shown in Table 4.6.

Because the matrices are symmetric, we only want to take each pairwise similarity into account once, and we ignore the trivial identity similarities. Hence we use the upper triangular versions of the matrices. Each upper triangular matrix U has $N = 43 \times 44/2$ entries above the main diagonal, corresponding to the 946 pairwise similarity values across all 44 words. We also calculate the Pearson correlation, r, between each of the matrices, U, and the baseline Leacock and Chodorow (upper triangular) similarity matrix, U_{LC}. The results can be found in Table 4.6.

Relation	M	F	r	Conf. Int.
With	McRae	15.53	0.4721	[0.467, 0.477]
	Wiki500	17.22	0.1553	[0.149, 0.162]
	Wiki100K	16.46	0.2084	[0.202, 0.215]
	BNC	16.44	0.3013	[0.296, 0.307]
	Wiki100K-BNC	16.26	0.2568	[0.251, 0.263]
Without	McRae	15.32	0.4780	[0.473, 0.483]
	Wiki500	16.76	0.2438	[0.238, 0.250]
	Wiki100K	15.77	0.2989	[0.293, 0.305]
	BNC	16.14	0.4170	[0.412, 0.422]
	Wiki100K-BNC	16.13	0.3109	[0.305, 0.317]

Table 4.6: Frobenious distances, Pearson correlation (r) results and confidence intervals between the Leacock and Chodorow WordNet M_{LC} matrix and the similarity matrices M from our best automatic extraction systems as reported in Table 4.4.

4.3.4 Human evaluation

We have already discussed many of the issues associated with employing the McRae/ESSLLI norms as our only point of comparison, and although the semantic similarity evaluations from the previous sections are indicative of whether we are going in the right direction, they are not absolute—our goal is, after all, to extract conceptual property norm-like information, not predict concept similarity. We therefore finally turn to human evaluation: it is arguably the ultimate arbiter of whether the triples we are extracting are indeed correct. Although some properties may not be easily verbalisable or might not come to mind when people list properties during a property norming study, humans can still evaluate whether or not a given property is true with relative

ease. That said, if a property truly cannot be verbalised then it will be decidedly absent in any corpus we use.

From the 44 concepts appearing in the ESSLLI set, we chose a selection of 15 upon which to carry out a human evaluation. When selecting the concepts from the 44 candidate concepts, we first excluded three of them: ***snail*** (as it had only 9 properties listed in the McRae set), ***onions*** (because it appeared in its plural form in the McRae set but as singular in the ESSLLI set) and ***truck*** (because this is known as 'lorry' in British English, the dominant dialect of the BNC). The remaining 41 concepts had already been classified into ten superordinate categories (e.g., 'animal') for unrelated psycholinguistic research, and we selected 15 concepts proportionally and at random from these superordinate categories. The selected concepts were: ***car***, ***cup***, ***duck***, ***hammer***, ***kettle***, ***knife***, ***lettuce***, ***lion***, ***motorcycle***, ***penguin***, ***pig***, ***pineapple***, ***potato***, ***screwdriver*** and ***turtle***.

Four native English-speaking judges evaluated the validity of the extracted relation/feature pairs. They were asked to choose between four possibilities for each triple: *correct* (c) when the triple represented a correct, valid, property; *plausible* (p) when the triple was plausible in a specic set of circumstances and/or was correct but very general; *wrong but related* (r) when the triple was wrong, but there existed some kind of relationship between the concept and the relation and/or feature; or *wrong* (w) when the triple was simply incorrect.

The human evaluation was executed across output from the four corpora (the three initial corpora, plus the combined Wiki100K/BNC corpus) and across all 300 triples (15 concepts $\times$ 20 triples) for each corpus. As there were shared triples across these corpora, each distinct triple was only evaluated once. The judges were unaware of the purpose of the study, and the evaluation was done blind with regard to the source extraction set for each triple (thus making a deliberate bias towards any one of the extraction sets impossible).

Although we asked the annotators to allocate each of the triples to one of four categories (*correct*, *plausible*, *wrong but related*, and *wrong*), we did this specifically to obtain more data for performing qualitative error-analysis of the system for future improvements, as well as to facilitate interpretation of the judgements themselves. The full instructions given to our participants are included in Appendix B, Sections B.1 and B.2. However, given the subjective nature of the judgements for the purposes of measuring inter-annotator agreement, we consider all triples judged as *correct* or *plausible* merely to be *correct* (since, given the above definition of *plausible*, these triples are indeed correct, even if only in general or in a specific set of circumstances), and all those marked as *wrong but related* or *wrong* to be *incorrect*. We measure the degree of inter-annotator agreement based on whether a triple is judged to be *correct* or *incorrect* by the four

judges. We use Fleiss' method (Fleiss, 1971) to calculate Kappa scores (Cohen, 1960) between the four annotators. Detailed agreement results when including the relation can be seen in Table 4.7, with a highest Kappa score of 0.427 ('moderate' agreement according to the labels assigned to various Kappa ranges by Landis and Koch (1977)) for the Wiki100K corpus. The corresponding set of results when excluding the relation terms can be found in Table 4.8. Here, the Kappa scores are slightly higher across the corpora, which is perhaps somewhat surprising since there is less information for the human judges to base their decision on, and hence one might expect them to be more likely to disagree. Throughout we can see that on average at least half of those triples marked as incorrect by the ESSLLI evaluation (both when including and excluding the relation) are considered to be *correct* or *plausible* by the judges. The judgments given for two concepts can be found in Table 4.9.

If we examine the generated triples and the human judgements in detail, we can see that for ***car***, the vast majority of the output is deemed correct by humans; the cases where there is disagreement often point to a property which is subjectively linked to the concept at hand (e.g., ***car*** *have* **crime**). On the whole the properties extracted are indisputably associated in some way with the concept at hand. For the concept ***penguin*** there is more disagreement between the annotators; this may be because some rather technical terms have been extracted from the corpus (our system has extracted the names of five separate species of penguin) and it is a subjective question as to the extent to which these species are features or conceptual properties of the concept ***penguin***. Furthermore, the fact that not one of the extracted features (many of which have been deemed correct or plausible by the human judges) appear in the ESSLLI gold standard again demonstrates the issues associated with that particular evaluation methodology.

Corpus		Judge A	B	C	D	Avg	Full agreement	Kappa
Wiki500	c / p	226	154	194	197	192.75	150 (50.0%)	0.401
	r / w	74	146	106	103	107.25		
Wiki100K	c / p	217	162	184	208	192.75	152 (50.7%)	0.427
	r / w	83	138	116	92	107.25		
BNC	c / p	231	175	208	235	212.25	181 (60.3%)	0.361
	r / w	69	125	92	65	87.75		
BNC-WIKI100K	c / p	237	185	208	229	214.75	168 (56.0%)	0.414
	r / w	63	115	92	71	85.25		

Table 4.7: Inter-annotator agreement for the four corpora, evaluating the best system with the relation included. 'Full agreement' corresponds to the number of times all four annotators gave the same rating (i.e., either c/p or r/w).

Corpus		Judge A	Judge B	Judge C	Judge D	Avg	Full agreement	Kappa
Wiki500	c / p	222	219	175	201	204.25	168 (56.0%)	0.444
	r / w	78	81	125	99	95.75		
Wiki100K	c / p	226	233	192	222	218.25	188 (62.7%)	0.486
	r / w	74	67	108	78	81.75		
BNC	c / p	208	206	195	201	202.5	194 (64.7%)	0.572
	r / w	92	94	105	99	97.5		
BNC-WIKI100K	c / p	232	236	217	235	230	207 (69.0%)	0.531
	r / w	68	64	83	65	70		

Table 4.8: Inter-annotator agreement for the four corpora, evaluating the best system, but excluding the relation. 'Full agreement' corresponds to the number of times all four annotators gave the same rating (i.e., either c/p or r/w).

Combining McRae and human evaluation

Given that we have collected human-evaluation data, it might also be instructive to assess—using the human ratings—how the system is performing based only on these 15 concepts. To do this, we calculate the precision for the top twenty features for each of the concepts from the sets (we are unable to calculate recall and F-scores because there is no upper bound on the number of triples which the human judges could deem correct). Each triple is judged as correct if and only if it is marked as either plausible or correct by *all* the judges. Results are shown in Table 4.10.

Since this evaluation is across only a relatively small number of concepts, it should not be interpreted as the full picture of how our system is performing. However, it does signal the extent to which patently incorrect triples are appearing. The results indicate that just under half of the triples returned are correct, and when evaluating on features alone, it is possible to achieve precision scores of over 60%, significantly outperforming our best-reported ESSLLI precision of 15%.

4.4 Discussion

In this experiment, we created a new automatic extraction system which employed a relatively small set of rules to extract candidate concept-relation-feature triples. We also introduced a new entropy-based measure for gauging the strength of a candidate triple based on the number of rules yielding that triple. We reweighted our system's candidate features in a novel way, using a number of statistical and semantic mea-

car	**Judge**				***penguin***	**Judge**			
	A	**B**	**C**	**D**		**A**	**B**	**C**	**D**
can be **motor**	c	c	c	c	*can be* **king**	c	c	w	c
can be **sport**	c	c	c	p	*be* **mascot**	c	p	p	p
have **crash**	c	c	p	c	*be* **species**	c	c	c	c
have **park**	r	p	c	c	*be* **game**	p	p	w	p
have **accident**	c	c	p	c	*can be* **adelie**	c	w	w	w
can be **electric**	c	c	c	c	*be* **character**	p	p	p	c
be **vehicle** (✓)	c	c	c	c	*can be* **young**	c	c	c	c
have **door** (✓)	c	c	c	c	*can be* **emperor**	c	c	c	c
can be **passenger**	c	c	c	c	*have* **book**	c	r	p	c
do **drive**	p	c	c	c	*can be* **african**	w	c	w	w
do **run**	c	c	p	c	*be* **large**	c	p	c	p
have **parking**	p	p	c	c	*be* **book**	c	r	p	c
can be **racing**	c	c	c	p	*be* **hoax**	p	w	w	p
have **driver**	c	c	c	c	*can be* **male**	c	c	c	c
can be **private**	c	c	c	c	*be* **adelie**	p	c	r	w
can be **small**	c	c	c	c	*be* **tall**	c	p	c	p
have **engine** (✓)	c	c	c	c	*can be* **giant**	c	c	c	r
can be **fast**	c	c	c	c	*be* **seal**	p	r	r	r
have **crime**	r	w	p	c	*name* **humboldt**	c	r	w	r
have **racing**	c	r	c	c	*do* **fly**	w	p	w	w

Table 4.9: Judgements for the ordered top twenty triples for two concepts from our best system output. A "✓" indicates that the triple is correct according to the ESSLLI evaluation set.

sures and compared how encyclopedic and general corpora (Wikipedia and the BNC) produce different 'types' of triples. Our system also aimed to extract behavioural features (marked by *do* relations) specifically exhibited by the concept under consideration, rather than activities merely associated with the concept at hand, an issue which previous systems have not broached. The evaluation analyses demonstrate consistently comparable performance with respect to previous work in the same domain and we offered a comprehensive evaluation of our system: we compared it directly with a gold standard, we asked humans to evaluate its output manually and we also measured our system's capacity for predicting both WordNet and human-rated similarities between concepts.

Our work examined the relative capacity of four distinct metrics to upweight human-like features/relations. The results indicated that two of these—the entropy of a triple, calculated from the probability mass across rules which generate it, and the semantic reweighting factor—offer improvements when evaluating against features alone and

	Relation	
Corpus	**With**	**Without**
Wiki500	0.3733	0.4500
Wiki100K	0.3867	0.5233
BNC	0.4967	0.4900
Wiki100K-BNC	0.4767	0.5967

Table 4.10: Precision scores for the top twenty triples from our automatic extraction system when evaluating the human judgements.

features with their relations. The two other measures (pointwise mutual information and log-likelihood ratio) offered less marked improvements; however, both did contribute to certain 'best' systems, depending on the corpus employed.

These results indicate that we have taken a significant step in the right direction—notwithstanding the various evaluation issues which we have already discussed, the results indicate solid (albeit slightly inferior) performance when compared to the 'best-possible' pilot method. We have shown a high level of accuracy on the output when judged by humans (almost 60% precision when judging on features alone and 50% correct or plausible features when including relations), and the semantic vector comparison indicates that our best system is not trailing all that far behind the McRae norms themselves when predicting human-judged semantic similarity of concepts. Given the issues we have outlined, perfect accuracy for this task against the gold standard evaluation set is unlikely.

We also note that when comparing our system with that of Baroni et al. (2009), using their evaluation criteria (i.e., calculating precision and recall on the top ten features only) our system (using the combined BNC/Wikipedia corpus and the reweighting parameters listed in Table 4.3) produces a best F-score of 0.207—their best F-score is 0.239—which we believe is a strong result, given the propensity for the evaluation to yield false negatives, and the fact that our system was optimised against the top twenty features rather than the top ten. It is also important to note that Baroni et al.'s method falls short of explicitly listing the relationships between the concepts and features it extracts, while our method was specifically designed for unambiguous relation extraction; our system is ambitious in attempting this. We believe this experiment forms an important step towards accomplishing this highly challenging task.

Chapter 5

Semi-supervised learning

In the previous chapter, we demonstrated that the use of directional grammatical relation patterns and part of speech information derived from parsed corpus-data can be beneficial in extracting candidate concept-relation-feature triples. In our first experiments, we generated rules over GR patterns manually, but we could also take a more generalised approach to this—for example, by examining the possibility of automating the rule-learning process to remove its manual element. This is what we hope to achieve in this chapter: employing semi-supervised training techniques to automatically acquire the rules. Such machine learning techniques have offered state of the art performance for many NLP tasks. In doing so, we hope to move beyond our reliance on manually crafted resources such as WordNet and hand-made rules, and head towards a more generally applicable approach which requires much less human-annotated data.

We propose to take a similar approach to that taken by Mintz et al. (2009) (see Section 2.3.3), who created a relation classifier using a paradigm called 'distant supervision' which assumed that "if two entities participate in a relation, any sentence that contains those two entities might express that relation." Our system will differ from theirs in that it can use any combination of lexical and syntactic attributes, implicit and explicit, obtained from the path linking the concept to the feature to generate a rule. We hope to empirically derive an optimal set of attribute-based patterns by using a portion of the known property norms as a training set to teach the system which patterns of GR-POS graph paths typically indicate plausible properties/triples. As we do not have as large a set of known relations (Mintz et al. (2009) trained on 900,000 held-out Freebase relations) we will probably not have the luxury of only matching on exactly identical features; we will need to lemmatise our training and test data.

In summary, our proposed system works by searching dependency-parsed corpora for those sentences containing concept and feature terms which are also found in a

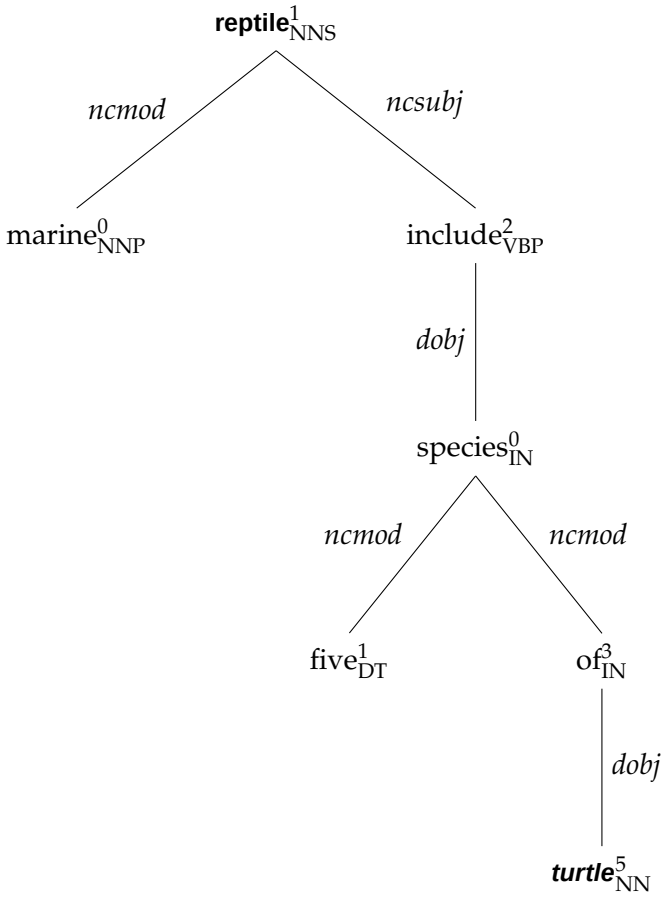

Figure 5.1: A C&C-derived GR-POS graph for the sentence *Marine reptiles include five species of turtle*.

McRae norm-derived training set of properties. For these sentences, the system generates grammatical relation/part of speech structural attributes and applies support vector machines (SVMs) to learn sets of attributes likely to indicate the instantiation of a property in a sentence. These learned patterns of salient attributes are then applied to a corpus to derive new properties for unseen concepts.

This chapter contains work from our published paper *Semi-supervised learning for automatic conceptual property extraction* (Kelly et al., 2012).

5.1 Data

5.1.1 Recoded norms

As before, these experiments use the British English version of the McRae norms. Given their provenance, the properties found in property norms are free-form. To sim-

plify our task we must, as before, apply a more rigid representation to the properties we already have and to those we aim to seek.

We again wish to delineate each property into a ***concept*** *relation* **feature** triple to render our task one of finding valid *relation* **feature** pairs given a particular ***concept***. This recoding makes this task more well-defined and also makes evaluation of our method more comparable to previous and related work. Therefore we will again recode the free-form McRae properties into relation-classes and features which will be usable for our learning algorithm. As we will be matching the features from these properties with individual words in the training corpus it is essential that the features we generate contain only one lemmatised word. However in contrast to previous work, the relations will act merely as labels for the relationship described (they do not need to occur in the sentences we are training from) and therefore need only be single-string relations. This allows the inclusion of prepositional verbs as distinct relations. This is something which has not been attempted in previous work, but which can be semantically significant (e.g., the relations *used-in*, *used-for* and *used-by* have dissimilar meanings).

We apply the following sequential multi-step process to the set of free-form properties to distill them to triples of the form ***concept*** *relation* **feature**, where *relation* can be a multi-word string and **feature** is a single word:

1. Translation of implicit properties to their correct relations (e.g., *pig an animal* → *pig is an animal*).

2. Removal of indefinite and definite articles.

3. Behavioural properties become 'does' properties (e.g., *turtle beh eats* → *turtle does eats*).

4. Negative properties given their own relation classes (e.g., *turkey does cannot fly* → *turkey doesnt fly*).

5. All numbers are translated to named cardinals (e.g., *spider has 8 legs* → *spider has eight legs*).

6. Some of the norms already contained synonymous terms: these were split into separate triples for each synonym (e.g., *pepper tastes hot/spicy* → *pepper tastes hot* and *pepper tastes spicy*).

7. Prepositional verbs were translated to one-word, hyphenated strings (e.g., *made of* → *made-of*).

turtle		***bowl***	
has a shell	25	is round	19
lays eggs	16	used for eating	12
swims	15	used for soup	11
is green	14	used for food	11
lives in water	14	used for liquids	10
is slow	13	used for eating cereal	10
an animal	11	made of plastic	8
walks	10	used for holding things	7
walks slowly	10	is curved	7
has 4 legs	9	found in kitchens	7

Table 5.1: Top ten properties from McRae norms with production frequencies for ***turtle*** and ***bowl***.

8. Properties with present participles as the penultimate word were split into one including the verb as the feature and one including it in the relation (e.g., *envelope used for sending letters* → *envelope used-for-sending letters* and *envelope used-for sending*).

9. Any remaining multi-word properties were split with the first term after the concept acting as the relation (e.g., *bull has ring in its nose* → *bull has ring*, *bull has in*, *bull has its* and *bull has nose*).

10. All remaining stop-words were removed; properties ending in stop-words (e.g., *bull has in* and *bull has its*) were removed completely.

This yields 7,518 property-triples with 254 distinct relations and an average of 14.7 triples per concept. Some sample properties for two concepts, ***turtle*** and ***bowl***, are listed in Table 5.1, and in Table 5.2 we list examples of recoded triples for the same two concepts.

5.1.2 Corpora

We employ two corpora for these experiments: the full text of Wikipedia (distinct from the previous Wiki100K corpus) and the UKWAC corpus (Ferraresi et al., 2008).

Our Wikipedia corpus is based on a September 2009 version of English-language Wikipedia and contains the vast majority of Wikipedia articles; around 1.84 million articles in total (>1bn words).

Our UKWAC corpus is an English-language corpus (>2bn words) obtained by crawling the `.uk` internet domain. UKWAC is a source of general text and, like Wiki-

turtle	***bowl***
does **swims**	*different* **colours**
does **walks**	*found-in* **kitchens**
has **four**	*is* **curved**
has **head**	*is* **round**
has **legs**	*made-of* **ceramic**
has **shell**	*made-of* **plastic**
is **amphibian**	*requires* **spoon**
is **animal**	*used-for* **eating**
is **green**	*used-for-eating* **cereal**
is **hard**	*used-for* **food**
is **reptile**	*used-for* **holding**
is **slow**	*used-for-holding* **things**
is **small**	*used-for* **liquids**
lays **eggs**	*used-for* **mixing**
lives-in **water**	*used-for* **soup**
lives-on **land**	
walks **slowly**	

Table 5.2: Recoded triples for ***turtle*** and ***bowl***.

pedia, is publicly available. We decided to use UKWAC instead of the BNC as a 'general text' corpus primarily due to its much larger size; it is around twenty times the size of the BNC. We believe this will increase its potential to further improve our extraction, especially for concepts and features with lower frequency.

5.1.3 Parser

We again use the C&C POS tagger and parser (Curran and Clark, 2003; Clark and Curran, 2007a) to parse both corpora, as we will employ both GR and POS information in our learning method. To accelerate this stage, we process only sentences containing a form (e.g., singular/plural) of one of the training/testing concepts, lemmatising each word using the WordNet NLTK lemmatiser (Bird, 2006). Parsing the corpora yields around 10Gb and 12Gb of text data for UKWAC and Wikipedia respectively.

5.2 Method

Machine learning, whether supervised or unsupervised, is an essential tool for NLP researchers. Highly developed supervised training techniques have offered state of the art performance for many NLP tasks. However, they are constrained by the (usually limited) availability of annotated data and the relatively high cost of obtaining

more—this is especially true in our case, as obtaining more property norms would be an expensive and time-consuming task. Unsupervised techniques have found applications in many parts of NLP (e.g., grammar induction, word-alignment for bilingual translation) and do not suffer from the same limits on data resources; however, unsupervised learning is much harder than supervised learning and is not always able to produce the consistently strong predictions required of an NLP system. Due to the relative rarity of property norm-like statements in corpora, we view that our task would be extremely difficult to perform without offering our system some form of prior knowledge as to what typically constitutes a property norm. Therefore, we aim to leverage both labelled and unlabelled data to improve performance in our system. Several papers have shown promising results with semi-supervised learning (e.g., tagging and parsing) and we are hopeful that we too can use these techniques to make progress on our task.

We will use support vector machines to learn lexico-syntactic patterns in the corpora corresponding to known properties in order to find new properties. Training an SVM requires a labelled training set. To generate this set we harness already-known concepts/features (and their relationships) from the McRae norms to find instantiations of said relationships within the corpora. We use parsed sentence information from the corpora to create a set of attributes describing each relationship, our learning patterns. In doing so, we assume that across those sentences containing a concept/feature pair found in the McRae norms there will be a set of consistent lexico-syntactic patterns which indicate the same relationship as that linking the pair in the norms.

In summary, we employ a subset of known properties from the McRae norms in the form of ***concept*** *relation* **feature** triples to iterate over the chosen corpora, parsing each concept-containing sentence to yield GR and POS information from which we can create a GR-POS graph relating the two. Then for each triple, we find any/all paths through the graph which link the ***concept*** to its **feature** and use the corresponding *relation* to label this path. We collect descriptive information about the path in the form of attributes describing it (e.g., path nodes, labels, length) to create a training pattern specific to that ***concept*** *relation* **feature** triple and sentence. It is these lists of attributes (and their *relation* labels) which we employ as the labelled training set and as input for the SVM.

5.2.1 Support vector machines

We use SVMs (Cortes and Vapnik, 1995) for our experiments as they have been used for a variety of tasks in NLP (e.g., including text-categorisation (Joachims, 1998), part of

speech tagging (Giménez and Marquez, 2004) and named-entity recognition (Kazama et al., 2002)) and their properties are well understood. Semi-supervised learning offers a flexible technique for leveraging small amounts of labelled data to derive information from unlabelled datasets/corpora and allows us to guide the extraction towards our desired 'common sense' output. We will demonstrate that this system's performance exceeds that of our previous systems and that of Reverb (Etzioni et al., 2011), another large-scale extraction system. This experiment is, as far as we are aware, the first work to employ semi-supervised learning for this task.

In their canonical form, SVMs are non-probabilistic binary linear classifiers which take a set of input data and predict, for each given input, which of two possible classes it corresponds to. This works by plotting training data points in a high-dimensional space and separating them with a hyperplane which has the largest distance (or 'margin') to the nearest training data points of each class. This plane is subsequently used to classify unseen data points.

In this case, there are more than two possible relation-labels to learn for the input patterns. Ours is hence a multi-class classification task. Crammer and Singer (2002) generalised the notion of margin in SVMs to the multi-class context. They used this notion to recast the multi-class SVM classification task to multiple constrained optimisation problems of reduced size and presented a fixed point algorithm capable of solving such reduced problems. This technique is preferable to the computationally expensive alternative of solving multiple independent binary SVM classification tasks.

The following experiments make use of the SVM Light Multiclass (v. 2.20) software (Joachims, 1999). Joachims' software, written in the C programming language, has been widely used to implement SVMs (Vinokourov et al., 2003; Godbole et al., 2002) and contains an implementation of Crammer and Singer's multiclass classification algorithm.

5.2.2 Attribute selection

Previous approaches to our task (and our previous two experiments) have made use of lexical, syntactic and semantic information for extraction. In this experiment we hope to avoid the use of manually created semantic resources, relying only on lexical and syntactic attributes for the learning stage (i.e., the GR-POS paths described earlier).

A table of all the categories of attributes we extract for each GR-POS path are in Table 5.3, together with attributes from the path linking ***turtle*** and **reptile** in the sentence:

Marine reptiles include five species of turtle.

The GR-POS graph for this example sentence can be found in Figure 5.1. The R and

Attribute category	Example attribute(s)
GR path-length	`LEN`
lemmatised anchor node	`LEM=turtle`
POS of anchor node	`POS=NN`

GR path labels from anchor (indexed)	`GR1=dobjR`
	`GR2=ncmodR`
	`GR3=dobjR`
	`GR4=ncsubjN`

GR path labels from target (indexed)	`GR1=ncsubjR`
	`GR2=dobjN`
	`GR3=ncmodN`
	`GR4=dobjN`

POS of path nodes from anchor (indexed)	`POS1=IN`
	`POS2=NNS`
	`POS3=VBP`
	`POS4=NNS`

POS of path nodes from target (indexed)	`POS1=NNS`
	`POS2=VBP`
	`POS3=NNS`
	`POS4=IN`

lemmatised path nodes (bag of words)	`LEM=include`
	`LEM=species`
	`LEM=of`

POS of all path nodes (set)	`POS=IN`
	`POS=NNS`
	`POS=VBP`

Relation verbs	N/A

GR path labels (set)	`GR=dobjR`
	`GR=ncmodN`
	`GR=ncsubjN`

lemmatised target node	`LEM=reptile`
POS of target node	`POS=NNS`

Table 5.3: An example vector for an instance of the relation-label *is*. The attributes are distinguished from one another by their attribute category. Relation verbs only appear in the verb-augmented vector-type and no such verbs appear in our example sentence, so this category of attribute is empty in this table. All attributes in the table will receive the value `1.0` except the `LEN` attribute which will have the value `0.2` (the reciprocal of the path length, 5).

`N` labels appended to the GRs distinguish between the two possible directionalities of that particular relation in the POS-GR graph.

We run the experiments with two vector-types which we call our 'verb-augmented' and 'non-augmented' vector-types. The sets are identical except the verb-augmented

vector-type will also contain an additional attribute category containing an attribute for every instance of a relation verb (i.e., a verb which is found in the training set of relations, e.g., *become*, *cause*, *taste*, *use*, *have* and so on) in the lexical path. We do this to ascertain whether this additional verb-information might be more informative to our system when learning relations (which tend to be composed of verbs).

We considered allocating an *unknownrel* relation label to those sets of attributes corresponding to paths through the GR-POS graph which did *not* link the concept to a feature found in the training data; however an initial analysis indicated the SVM model would merely assign every pattern we tested to the *unknownrel* relation. Therefore we used only positive instances in the training pattern data.

Relation	Vector	Corpus	β_{LL}	β_{PMI}	β_{SVM}	Prec.	Recall	F
With	Non	Wikipedia	0.05	0.00	1.00	0.1199	0.1732	0.1394
		UKWAC	0.05	0.00	1.00	0.1126	0.1633	0.1312
		Combined	0.05	0.00	0.65	0.1241	0.1808	0.1449
	Verb	Wikipedia	0.05	0.00	1.00	0.1215	0.1747	0.1410
		UKWAC	0.05	0.00	1.00	0.1190	0.1724	0.1387
		Combined	**0.05**	**0.00**	**0.70**	**0.1281**	**0.1860**	**0.1494**
Without	Non	Wikipedia	0.30	0.00	1.00	0.2214	0.3197	0.2564
		UKWAC	0.10	0.05	0.60	0.2279	0.3330	0.2664
		Combined	0.35	0.00	0.75	0.2422	0.3533	0.2829
	Verb	Wikipedia	0.20	0.00	0.65	0.2217	0.3202	0.2568
		UKWAC	0.30	0.00	0.95	0.2326	0.3400	0.2720
		Combined	**0.40**	**0.05**	**1.00**	**0.2444**	**0.3577**	**0.2859**

Table 5.4: Parameter estimation for the semi-supervised learning system, using our verb-augmented ('Verb') and non-verb-augmented ('Non') vector-types, across the two corpora and the combined corpus.

Hence, we cycle through all training concept-feature pairs, finding sentences containing both terms. For each such sentence, the system generates the attributes from the GR-POS path linking the concept to the feature (the linking-path) to create a pattern for that pair, in the form of a relation-labelled vector containing real-valued attributes. The system assigns `1.0` to all attributes occurring in a given path and the `LEN` value receives the reciprocal of the path-length. All other possible attributes are assigned the value `0.0`. Each linking-path is collected into a *relation*-labelled, sparse vector in this manner. In the larger UKWAC corpus this corresponds to over 29 million unique attributes across all found linking-paths (this figure corresponds to the dimensionality of the vectors). We then pass all vectors to the learning module[1] of SVM Light to generate a learned model across all training concepts.

[1]Using a regularisation parameter (`C`) value of 1.0 and default parameters otherwise. See Joachims (1999) and Tsochantaridis et al. (2004) for details.

5.2.3 Extracting candidate patterns

Having trained the model, we must now find potential features and relations for our test concepts in the corpora. We again only examine sentences which contain at least one of the test concepts. Furthermore, to avoid a combinatorial explosion of possible paths rooted at those concepts we only permit as candidates those paths whose anchor node is a singular or plural noun and whose target node is either a singular/plural noun or adjective. This filtering corresponds to choosing patterns containing one of the three most frequent anchor node POS tags (`NN`, `NNS` and `NNP`) and target node POS tags (`NN`, `JJ` and `NNS`) found during the training stage. These candidate patterns constitute 92.6% and 87.7% of all the vectors, respectively, from our training set of patterns (on the UKWAC corpus). This pattern pre-selection allows us to immediately ignore paths which, despite being rooted at a test concept, are unlikely to contain property norm-like information.

5.2.4 Generating and ranking triples

We next classify the test concepts' candidate patterns using the learned model. SVM Light assigns each pattern a relation-class from the training set and outputs the values of the decision functions from the learned model when applied to that particular pattern. The sign of these values indicates the binary decision function choice, and their magnitude acts as a measure of confidence. We want those vectors which the model was most confident in across all decision functions, so we take the sum of the absolute values of the decision values to generate a pattern score for each vector/relation-label. From these patterns we derive an output set of triples where the concept and feature of a triple correspond to the anchor and target nodes of its pattern and the relation corresponds to the pattern's relation-label. Identical triples from differing patterns had their pattern scores summed to give a final SVM score for that triple.

5.2.5 Calculating triple scores

A brief qualitative evaluation of our system's output indicates that although the higher-ranked (by SVM score) features and relations are, for the most part, quite sensible, there are some obvious output errors (e.g., non-dictionary strings or verbs appearing as features). Therefore we restrict the features to those which appear as nouns or adjectives in WordNet and exclude features containing an NLTK (Bird, 2006) corpus stop-word. Despite these exclusions, some general (and therefore less informative) relation/feature combinations (e.g., *is* **good**, *is* **new**) still rank highly.

To mitigate this, we again extract both log-likelihood (LL) and pointwise mutual information (PMI) scores (see Section 4.2.2) for each concept/feature pair to assess the relative saliency of each extracted feature, with a view to downweighting common but less interesting features. To speed up this and later stages, we calculate both statistics for the top 1,000 triples extracted for each concept only.

We calculate an overall score for a triple, t, by a weighted combination of the triple's SVM, PMI and LL scores using the following formula:

$$\text{score}(t) = \beta_{\text{PMI}} \cdot \text{PMI}(t) + \beta_{\text{LL}} \cdot \text{LL}(t) + \beta_{\text{SVM}} \cdot \text{SVM}(t) \tag{5.1}$$

where the PMI, SVM and LL scores are normalised so they are in the range [0, 1]. The relative β weights therefore again give an estimate of the three measures' importance relative to one another and allow us to gauge which combination of these scores is optimal.

We employ ten-fold cross-validation to derive optimal SVM, LL and PMI β parameters for our final system. To begin, we exclude the 44 ESSLLI concepts from the set of 510 to use in our final system testing and split the remaining 466 concepts randomly and evenly into ten folds. We apply the training steps above to nine of the folds, generating predictions for the single held-out fold. We repeat this for all ten folds, yielding relations and features with SVM, LL and PMI scores for the full set of 466 training concepts for each of the corpora.

We again want to ascertain the extent to which the output from both the corpora could be combined to improve results, balancing the encyclopedic but somewhat specific nature of Wikipedia with the generality and breadth of the UKWAC corpus. Therefore we also combine the output by summing individual SVM scores of each triple from both corpora to yield a 'combined' SVM score. PMI and LL scores are then calculated as usual from this combined set of triples.

As before, we vary the β values from our scoring equation (Equation 5.1) in the range [0,1] (interval 0.05) and compare the top twenty triples for each concept directly against the held-out training set. The best F-scores and their corresponding β values (evaluating on full triples and concept-feature pairs alone) are in Table 5.4. We can see that the best results employ the verb-augmented vector-type and the combined corpus, with best F-scores of 0.2859 when ignoring the relation term and 0.1494 when including it. The main difference between these two results is the relative contribution of the reweighting factors: the SVM score is the most important overall, but the LL and PMI scores come into play when evaluating without the relation. This could be explained by the fact that the PMI and LL scores do not use any relation terms in their calculations, while the SVM scores were derived from our learning model which was

Relation	System	Prec.	Recall	F
With	Pilot	0.1102	0.2210	0.1471
	ReVerb	0.0431	0.0864	0.0576
	Wikipedia	0.1179	0.2365	0.1573
	UKWAC	0.1131	0.2272	0.1510
	Combined	0.1238	0.2493	0.1654
Without	Pilot	0.1943	0.3896	0.2593
	ReVerb	0.1142	0.2258	0.1514
	Wikipedia	0.2310	0.4627	0.3081
	UKWAC	0.2298	0.4611	0.3067
	Combined	0.2417	0.4847	0.3225

Table 5.5: Precision, Recall and F-scores across the three corpora on the ESSLLI set compared to our best pilot system results and the ReVerb system. The results are from the verb-augmented vector-type, using the β parameters highlighted in Table 5.4.

specifically optimising for correct relation selection.

5.3 Evaluation

5.3.1 Gold standard evaluation

We employ the ESSLLI set to test the final output. We use the two best systems (i.e., including and excluding the relation; highlighted in Table 5.4) to generate two sets of top twenty output triples for the 44 concepts. We then calculate precision, recall and F-scores for each against the synonym-expanded set.[2] Using this expanded set allows us to compare this work with our pilot system. We also compare with the top twenty output of the ReVerb system Etzioni et al. (2011) using their publicly available relations derived from the ClueWeb09 corpus, employing their normalised triples ranked by frequency. All sets of results are in Table 5.5. We note that even though our pilot system was optimised on the ESSLLI set to yield theoretical best-possible scores—we are evaluating 'blind'—our performance still shows an advance on those scores: the improvement on both sets when comparing the population of F-scores across all 44 concepts is statistically significant at the 0.5% level.[3]

[2]We note that we are still incorporating an upper bound for precision of 0.500 by comparing the top twenty output with the ten ESSLLI properties for each concept.

[3]Paired t-tests. With relation: $t = 3.524$, d.f.$= 43$, $p = 0.0010$. Without relation: $t = 3.503$, d.f.$= 43$, $p = 0.0011$.

5.3.2 Human-generated semantic similarity comparison

We also compare this system's output with the human-generated semantic similarity scores, using the same method as that described in Section 4.3.2. The results can be found in Table 5.6.

The first thing to notice is that, compared to our previous automatic extraction system, we have reduced the total number of distinct relation-feature combinations while the number of distinct features has increased. This seems plausible: we are limiting the relations to those found in the McRae minus ESSLLI training set (thereby limiting them in number). At the same time, our algorithm is able to generalise over the training examples to extract a greater diversity of features rather than being limited by a fixed number of rules for feature extraction. This means that the 'with relation' method has a comparable number of distinct triples to the McRae norms themselves (taking into account the 20% difference in size of the extracted property sets). The features-only method, on the other hand, has moved further away from the original norms, generating more features on average than our automatic extraction system.

Also of interest is the significant improvement in terms of correlation when considering this system's output with both relations and features included. The UKWAC and combined corpora both have correlations with the human data of around 0.70, which is not far off the McRae set's correlation (0.79). However, one could also argue that since we are deriving the relations directly from the McRae norms, this result is not all that surprising. Interestingly, the semi-supervised learning is not performing as well on the features-only evaluation as our previous system (correlation average of 0.42 compared to the previous average of 0.56).

Relation	V	D	r	**Conf. Int.**
With	McRae	410	0.7853	[0.691, 0.854]
	Wikipedia	492	0.6342	[0.492, 0.744]
	UKWAC	471	0.7010	[0.578, 0.793]
	Combined	495	0.6959	[0.571, 0.789]
Without	McRae	355	0.7874	[0.693, 0.855]
	Wikipedia	582	0.4409	[0.257, 0.594]
	UKWAC	587	0.4528	[0.271, 0.603]
	Combined	604	0.3655	[0.171, 0.532]

Table 5.6: Pearson correlation (r) results between the V_{Human} vector and the similarity vectors V (and their vector dimensionalities D) from our best semi-supervised learning systems as reported in Table 5.5.

5.3.3 WordNet semantic similarity comparison

We repeat the WordNet semantic similarity evaluation as described in Section 4.3.3. The results can be found in Table 5.7. In terms of the WordNet comparison, it is clear that our latest system has a much lower correlation with the semantic similarity scores across all the corpora, despite being 'closer' in terms of the Frobenious norm evaluation. Such low correlation scores indicate that the output from this system would not act as a good proxy for semantic similarity. This is a rather surprising result as it appears to contradict our findings from the previous section, where we had reasonably strong correlation with human similarity evaluations (albeit on a much smaller set of pairs).

However it is possible that the stronger correlation results for the same evaluation on our previous experiment may be, at least in part, due to the importance of the WordNet semantic clustering in the scoring of that system; as a consequence, the system structured and prioritised output triples in a way similar to that of the WordNet hierarchy. This present experiment, on the other hand, does not employ the WordNet ontology, which could explain the poorer correlation (albeit still positive) with its semantic similarity values.

Relation	***M***	***F***	***r***	**Conf. Int.**
With	McRae	15.53	0.4721	[0.467, 0.477]
	Wikipedia	14.35	0.1075	[0.101, 0.114]
	UKWAC	14.08	0.1241	[0.118, 0.130]
	Combined	13.87	0.1050	[0.099, 0.111]
Without	McRae	15.32	0.4780	[0.473, 0.483]
	Wikipedia	14.02	0.1316	[0.125, 0.138]
	UKWAC	12.96	0.1192	[0.113, 0.126]
	Combined	13.35	0.1251	[0.119, 0.131]

Table 5.7: Frobenious distances, Pearson correlation (r) results and confidence intervals between the Leacock and Chodorow WordNet M_{LC} matrix and the similarity matrices M from our best semi-supervised learning systems as reported in Table 5.5.

5.3.4 Human evaluation

As ever, our gold standard does not quite offer the full picture: it is possible that there are correct properties being generated which simply don't appear in the ESSLLI evaluation set.

Hence we again perform a human evaluation on 15 of the concepts. We asked two native English-speaking judges to annotate the output, using the same criteria as

	Judge			Judge	
turtle	**A**	**B**	***bowl***	**A**	**B**
is **green**	c	c	*is* **large**	p	p
is **small**	c	c	*used for* **food**	c	c
is **species**	c	c	*used for* **mixing**	c	c
is **marine**	c	c	*used for storing* **food**	c	c
used for **sea**	r	r	*used for storing* **soup**	r	r
is **animal**	c	c	*is* **ceramic**	c	c
is **many**	p	c	*is* **small**	p	p
has **shell**	c	c	*used for storing* **cereal**	r	r
is **large**	c	p	*used for storing* **spoon**	r	r
is **reptile**	c	c	*used for storing* **sugar**	p	c

Table 5.8: Our judges' assessments of the correctness of the top ten relation/feature pairs for two concepts extracted from our best system.

		Judge				
Relation		**A**	**B**	**Avg**	**Kappa**	**Agreements**
With	c / p	147	162	154.5	0.7421	261 (87%)
	r / w	153	138	145.5		
Without	c / p	226	235	230.5	0.5792	255 (85%)
	r / w	74	65	69.5		

Table 5.9: Inter-annotator agreement for our best system, both including and excluding the relation.

described in Section 4.3.4.

We executed the human evaluation on our two best systems (with and without relation terms). As there were shared triples and concept-feature pairs across the two output sets, each triple and pair was evaluated only once. The judges were aware of the purposes of the study but were blind to the source sets. Some example judgements are in Table 5.8.

The agreement results across all 15 concepts together with their Kappa coefficients (Cohen, 1960) are in Table 5.9. In this evaluation we again conflate the *correct/plausible* and *wrong but related/wrong* categories. These results indicate that our system is extracting correct or plausible triples 51.1% of the time (rising to 76.8% when considering features only), an improvement on our automatic extraction system. They also again demonstrate a marked discrepancy from the gold standard evaluation, further reflecting the necessity of human evaluation when assessing this particular task.

5.4 Discussion

In this chapter we have demonstrated that semi-supervised learning techniques can automatically learn lexico-syntactic patterns indicative of property norm-like relations and features. Using these patterns, our system extracts relevant and accurate properties from the parsed corpora and allows for multi-word relation labels, allowing greater semantic precision. The results clearly show that there are gains to be made through employing semi-supervised learning techniques to this task. We better the performance of both of our previous systems, even when evaluating against an unseen set of concepts, and our system does not use manually generated rules or WordNet-derived semantic information. Furthermore, human evaluation shows over half of the extracted properties are correct/plausible.

Chapter 6

Improving relation extraction

THE STRENGTHS OF OUR SYSTEMS so far lie mainly in their ability to extract features reasonably well. Extracting correct relations, on the other hand, seems to present more difficulties, as illustrated by our various evaluations. This indicates that it might be worthwhile restructuring the system so that it first extracts likely features, only later returning to the corpus to find probable relations for those features. In this final experiment, we aim to harness the strong results from our semi-supervised learning system in terms of its feature extraction, and additionally allow for unconstrained relation discovery.

Approaching the task in this way draws on some of the advantages of different aspects of our previous work and introduces new benefits as well:

1. Unlike in our previous semi-supervised learning experiment, we are no longer constrained by relations that appear solely in the McRae norms; this method allows for the extraction of any relation.

2. It allows us to break down the problem into its constituent parts (i.e., finding relevant features first, and then finding their salient relationships with the concept).

This chapter thus presents a strongly performing, minimally supervised technique for unconstrained relation and feature extraction. It contains work from our paper *Minimally supervised learning for unconstrained conceptual property extraction* (Kelly et al., 2013).

6.1 Data

6.1.1 Recoded norms

We use the same training set of norms as employed in the previous experiment (see Section 5.1.1).

6.1.2 Corpora

As in the previous experiment, we use the UKWAC corpus and the full Wikipedia corpus, as well as the combination of the two.

6.1.3 Parser

We again use the C&C-parsed versions of the corpora.

6.1.4 Chunking

For this experiment we also use chunked versions of the two corpora. Chunking is a technique which identifies the constituent blocks of a sentence (verb phrase, noun phrase, prepositional phrase, etc.). The primary advantage is that it is significantly faster than full parsing, yet the granularity of the produced sentence divisions is at a level which groups strongly syntactically linked terms together. This makes it easier to pick out the most important components of a relation whilst generalising over less important adjectives/adverbs in a sentence. For example, the sentence

The bear seemed to be very dangerous.

would be chunked to:

```
[NP The_DT bear_NN ] [VP seemed_VBD to_TO be_VB ]
  [ADJP very_RB dangerous_JJ ] ._.
```

From this we can see that by extracting the noun from the output's noun phrase (`NP`), the verb from the verb phrase (`VP`) and the adjective from the adjectival phrase (`ADJP`) we could extract the triple ***bear*** *be* **dangerous**. We therefore believe that using the output from chunking could be ideally suited to this subtask of relation extraction.

To chunk the corpora, we used the Apache OpenNLP 1.5 suite (Baldridge, 2005), using the Tokenizer, POS Tagger and Chunker tools. The various components of the suite were trained using models supplied with the OpenNLP package: the Tokenizer model

was trained on OpenNLP data; the POS Tagger model was trained using the Penn Treebank tag set and Ratnaparkhi's maximum entropy model (Ratnaparkhi, 1996); the chunker model was trained using data from the CoNLL-2000 shared task. Using these tools we were able to transform the corpora from plain-text English to chunked text with POS tags.

6.2 Method

Our method works in four main stages:

1. Feature derivation: we use similar techniques to the previous experiment to extract likely features from the training corpus, using a lightly supervised method.

2. Relation extraction: in parallel with the first stage, we select those sets of contiguous chunks in the corpus sentences a) which contain one of the target concepts, and b) whose labels match one of the training data-derived label patterns.

3. Relation selection: for each concept we choose the most promising features found in the first two stages, and then, using a backing-off technique, establish the most likely relation for each concept/feature pair. We also gather statistics related to the concept, feature and relation of each generated triple.

4. Reweighting: we use a linear combination of our various metrics to assign each triple a score and use a stochastic algorithm to determine the optimal parameters for that scoring scheme.

6.2.1 Feature derivation

In the first stage we focus on only extracting features which are relevant to the concepts at hand. If we can achieve this with reasonable accuracy, we will then have a promising set of features with which to anchor the relations: we believe that it will be much easier to find the relations between a concept and its feature than trying to find relation and feature at the same time.

As in the two previous experiments, we train our system using the 466 held-out non-ESSLLI concepts.

Machine learning attributes

We train the support vector machine in an identical manner to our previous experiment, as described in Section 5.2.2, however in addition to those attributes listed in

Table 5.3, we include two additional attribute categories: bigrams and concept/feature clusters. In other words, we also include all bigrams lying between the anchor and target, as well as two attributes corresponding to the semantic clusters of both the anchor and the target terms. As in Section 3.2.2, we use hierarchical clustering on WordNet to derive these clusters, with 50 clusters for the anchors (concepts) and 150 clusters for the targets (features). The additional attributes as would be applied to the turtle/reptile example sentence (see Section 5.2.2 and Table 5.3) are listed in Table 6.1.

Attribute category	**Example attribute(s)**
Bigrams	`BGM=marine_reptiles`
	`BGM=reptiles_include`
	`BGM=include_five`
	`BGM=five_species`
	`BGM=species_of`
	`BGM=of_turtle`
Anchor cluster ID	`CCID=34`
Target cluster ID	`FCID=6`

Table 6.1: Our new vector's additional attributes to those listed in Table 5.3 for the same instance of the relation-label *is*.

We hope that the introduction of these additional parameters will further guide the machine learning algorithm. We note that, although we are again using WordNet clusters, we have not calculated the semantic reweighting factor (see Section 3.2.2). We therefore expect this clustering to have much less of an impact on the final output than it has in our previous experiments.

Learning instances

In the previous experiment we ignored a large amount of potentially instructive training data. Specifically, we did not use those GR-POS paths in the corpus which did not terminate on one of the training features, nor did we employ those paths through sentences containing one of the concepts but none of the training features. It might therefore be worthwhile investigating the use of this 'negative' information.

Hence, instead of only using positive instances of relationships from the training data, we now employ as training data all grammatical relation paths linking one of the concepts to *any* potential target term within each sentence (i.e., a term satisfying the criteria listed in Section 5.2.3). This means that the size of the training set is 5.52 million instances for the Wikipedia corpus and 20.07 million instances for the UKWAC corpus. As we are unaware of the nature of the relationship between the vast majority of the anchor/target terms, we label these unknown training paths as *unknownrel*. Those paths

matching the McRae norms were still assigned their respective relations from those norms, however these now only form a very small proportion of the training data. By doing so we are effectively rendering the system only very lightly supervised: 6.8% of the UKWAC input and 8.7% of the Wikipedia input to the system is labelled with relations drawn from the McRae norms. The outcome of this is that every concept/feature pair that the SVM generates is joined by the *unknownrel* relation. This is not a problem, in that we intend only to use the feature output from this stage of the system, using the top 200 returned concept/feature pairs (and their SVM scores) as input to the next stage of the system.

To avoid memory issues associated with the sheer volume of training instances we employed two-fold cross-validation (rather than ten-fold, as in the previous experiment) to train over the 466 concepts.

6.2.2 Relation extraction

The underlying hypothesis of the relation extraction stage is that if we find sequences of chunks in the corpus sentences which are anchored at each end by a known ***concept*** and **feature** (from the previous stage), and those chunks' labels are the same as the chunk labels of the (chunked) property norms, then we will be able to use the chunk(s) between the anchors as the *relation* in the ***concept*** *relation* **feature** format.

Chunk pattern selection

To decide on what patterns of chunks would be likely to be indicative of property norm relations, we turned to the training set. We passed the full text of the non-ESSLLI McRae norms through the chunker, and manually examined the output to detect patterns which we could use when selecting chunks likely to indicate relations. For example, three property norms listed in the McRae training set are *mirror found in bedrooms*, *sofa is comfortable* and *trumpet used by blowing through*. Passing these to the chunker yields the following output:

1. `[NP mirror_NN ] [VP found_VBD ] [PP in_IN ] [NP bedrooms_NNS]`
2. `[NP sofa_NN ] [VP is_VBZ ] [ADJP comfortable_JJ ]`
3. `[NP trumpet_NN ] [VP used_VBD ] [PP by_IN ] [VP blowing_VBG ] [ADVP through_RB ]`

Each pair of square brackets encloses a chunk, and we call the first term between the brackets that chunk's label. We call a sequence of three chunks a three-chunk, a

Label pattern	Freq.	%
NP VP NP	2182	35.0
NP VP PP NP	2144	34.4
NP VP ADJP	1362	21.9
NP VP ADVP	271	4.4
NP PP NP	112	1.8
VP PP NP	87	1.4
Other	70	1.1

Table 6.2: Frequency counts for and relative proportions of the various combinations of chunk labels across the set of three- and four-chunks extracted from the training (non-ESSLLI) norms.

sequence of four chunks a four-chunk and so on. In this way the first item in the above list is a four-chunk labelled NP VP PP NP, the second a three-chunk labelled NP VP ADJP and so on. We applied this process to all property norms in the training set to yield chunk label-sets for every property.

In examining the output, we wanted to detect strong patterns indicating property norm-like phrases in text which we could harness for relation extraction. It was clear that the vast majority of the label-sets (91.9%) corresponded to three- and four-chunks. 6.3% of the output was in one- and two-chunks, however upon examination it appeared that a significant proportion of these 'chunks' contained errors (understandably so, given the isolated sentence fragments which we were offering as input to the chunker) and therefore would likely not be instructive when it came to deriving relations from them. The five- and six-chunks similarly only constituted a small proportion of the output (1.8%) and furthermore there was no strong pattern of chunking labels which we could obviously use without potentially introducing large amounts of noise to our method, for relatively little gain. We therefore elected to work only with three- and four-chunks. The breakdown of our returned label-sets for the three- and four-chunks can be found in Table 6.2.

Having established that we wish to use three- and four-chunks for the purposes of extracting relations, we now set about creating a ruleset for selecting sentence fragments (chunk sequences) which are similar in structure to the property norms. We decided to employ the first four most frequent label combinations to form our ruleset, as together these cover 95.6% of the three- and four-chunk label patterns generated from the training set.

We note that by using the NP VP PP NP-labelled four-chunks we are now also allowing the system to extract multi-word, prepositional verbs (e.g., *worn on*, *used for*) as potential relations. This is something which our previous relation extraction systems

have not attempted.

Chunk pre-selection

Having decided on our chunk label patterns, we now need to select those chunks which are most relevant to the relation extraction task. To do this we pass through the chunked corpus, generating sets of 3 and 4 sequential chunks and pre-selecting those which are relevant to the concepts. Our criterion for relevancy at this stage for the three- and four-chunks is that the final term contained within the first chunk, when lemmatised, must correspond to one of the training concepts.

In other words, we generate all possible three-chunks and select only those which meet the following criteria:

1. The first chunk must be labelled `NP` (noun-phrase).
2. The second chunk must be labelled `VP` (verb-phrase).
3. The third chunk must be labelled `NP` (noun-phrase), `ADVP` (adverbial phrase) or `ADJP` (adjectival phrase).

We also examine all possible four-chunks and select only those which meet the following criteria:

1. The first chunk must be labelled `NP` (noun-phrase).
2. The second chunk must be labelled `VP` (verb-phrase).
3. The third chunk must be labelled `PP` (prepositional-phrase).
4. The fourth chunk must be labelled `NP` (noun-phrase).

Chunk to triple conversion

Having pre-selected the chunks we now wish to generate triples from the chunk text. For three-chunks we do this by simply taking the final term in the first, second and third chunks and lemmatising each to give the ***concept***, *relation* and **feature** terms respectively. For four-chunks we follow the same process for the first and fourth chunks to yield the ***concept*** and **feature**. To extract the *relation* we take the final term of the second (`VP`) chunk and compound it with the final term of the third (`PP`) chunk to give the relation; the only exception to this is if the POS of the final term of the second chunk is `VBG`, in which case we lemmatise that term and compound it with the third chunk's final term. For example:

- `[NP Mirrors_NNS ] [VP are_VBP found_VBN ]`
  `[PP in_IN ] [NP the_DT bedroom_NN]` becomes ***mirror*** *found in* **bedroom**
- `[NP Most_JJS cats_NNS ] [VP have_VBP ]`
  `[NP furry_NN tails_NNS]` becomes ***cat*** *have* **tail**
- `[NP The_DT microwave_NN ] [VP was_VBD running_VBG ]`
  `[PP on_IN ] [NP electricity_NN]` becomes ***microwave*** *run on* **electricity**

We concede that this is a simplification, and won't necessarily always be a true reflection of the sentence's original meaning. It is, for example, possible for the final chunk to contain adjectives which modify the final noun which could either have importance from a conceptual representation perspective (e.g., features such as **long neck** for ***giraffe*** *has* **long neck**). It is also possible that the modifying portion of a chunk may be semantically significant and greatly alter the final term's meaning (e.g., a ***tea bag*** is quite different from a ***bag***). In future work, however, it should be possible to have more general chunk to triple extraction; we discuss this in the next chapter.

Example

The entire relation extraction process is best illustrated with an example. Consider the sentence:

The pan was removed from the heat while the oven continued to bake the main dish at 180 degrees.

This sentence contains three of the McRae concepts, ***pan***, ***oven*** and ***dish***. Chunking this sentence yields the following output:

```
[NP The_DT pan_NN ] [VP was_VBD removed_VBN ] [PP from_IN ]
  [NP the_DT heat_NN ] [SBAR while_IN ] [NP the_DT oven_NN ]
  [VP continued_VBD to_TO bake_VB ] [NP the_DT main_JJ dish_NN ]
  [PP at_IN ] [NP 180_CD degrees_NNS] ._.
```

As this chunk output has $n = 10$ chunks in total, it will generate a total of $n - 2$ three-chunks and $n - 3$ four-chunks giving a total of $2n - 5 = 15$ chunk-sets in total. The pre-selection stage for the three-chunks would immediately eliminate all but three chunk-sets (these are the only three-chunks which have, when lemmatised, the final term of the first chunk corresponding to one of the concepts):

1. `[NP The_DT pan_NN ] [VP was_VBD removed_VBN ]`
   `[PP from_IN ]`

2. `[NP the_DT oven_NN ] [VP continued_VBD to_TO bake_VB ] [NP the_DT main_JJ dish_NN ]`
3. `[NP the_DT main_JJ dish_NN ] [PP at_IN ] [NP 180_CD degrees_NNS]`

Similarly for the four-chunks, the pre-selection eliminates all but two of the chunk-sets, as none of the other four-chunks contain our target concepts in the final slot of the first chunk:

4. `[NP The_DT pan_NN ] [VP was_VBD removed_VBN ] [PP from_IN ] [NP the_DT heat_NN ]`
5. `[NP the_DT oven_NN ] [VP continued_VBD to_TO bake_VB ] [NP the_DT main_JJ dish_NN ] [PP at_IN ]`

We can then apply our chunk label pattern criteria outlined above, leaving only two chunk-sets: #2 (matching the `NP VP NP` pattern) and #4 (matching the `NP VP PP NP` pattern). We finally convert these two remaining chunk-sets to potential triples following our conversion process, yielding ***oven*** *bake* **dish** for the first chunk and ***pan*** *removed from* **heat** for the second.

6.2.3 Relation selection

The third stage of the system works by taking each ***concept*–feature** pair from both the SVM and chunking output, and finding the best relation for that pair from the chunking output to generate a triple. It also assigns to that triple a number of metrics relating to its constituent parts, their relative frequency and association scores.

We are still making the assumption that each ***concept*–feature** pair has one corresponding relation (this is demonstrably false in many cases, but we can view our task to be one of selection of the most appropriate *relation* for that concept/feature pair).

We call the set of extracted triples generated by Stage 2 T (with triples $(c, r, f) \in T$) and the set of all extracted relations from Stage 2, R. We call our set of concepts C.

For each concept, we also generate a final potential feature set, F_c, which, for a given concept, is the union of the top 200 features from Stage 1 (ranked by their SVM score) and the top 200 features from Stage 2 (ranked by frequency in the extracted relations, but excluding those features which appear once only).

We first define Concept Feature Frequency (CFF) to be the number of times a concept and feature co-occur across the extracted relations:

$$\mathrm{CFF}(c, f) = \sum_{r \in R} \mathrm{freq}(c, r, f) \tag{6.1}$$

We also calculate a Distinct Relation Score for each concept and feature, which we call $\text{DRS}(c, f)$:

$$\text{DRS}(c, f) = |D_{c,f}| \text{ where } D_{c,f} = \{r : (c, r, f) \in T\} \tag{6.2}$$

That is, the Distinct Relation Score measures the number of distinct relations linking c to f.

We next want to choose relations for the various ***concept*–feature** pairs, $(c, f) \in C \times F_c$. We do this using three steps:

Step 1

For each concept, c, and feature, f, we iterate through all relations relating to that pair and calculate an Exact Match Score:

$$\text{EMS}(c, f) = \max\{\text{freq}(c, r, f) : r \in R\} \tag{6.3}$$

If $\text{EMS}(c, f) > 0$ then we select as best relation, $\hat{r}$, the relation corresponding to that score. If there is more than one relation with the same score, then we choose the least common (i.e., that which has the lowest frequency across all relations). If $\text{EMS}(c, f) = 0$ then we leave $\hat{r}$ undefined.

Step 2

Our first step only retrieves a relation if there is an exact match amongst the relation extraction output. However, this is not always the case, and we therefore need to derive a way to generate relations which we do not have exact matches for.

To achieve this we decide to take a split approach; given a particular concept, c and feature, f, we calculate separate probabilities across all the relations of c occurring with each relation, and of f occurring with each relation. We can then calculate for each relation r a combined score for the combination of c, r and f by multiplying the constituent probabilities together. The Pairwise Combination Score is defined as:

$$p(c, r) = \sum_{f \in F} \frac{\text{freq}(c, r, f)}{\text{freq}(c) \times \text{freq}(r)} \tag{6.4a}$$

$$p(r, f) = \sum_{c \in C} \frac{\text{freq}(c, r, f)}{\text{freq}(r) \times \text{freq}(f)} \tag{6.4b}$$

$$\text{PCS}(c,f) = \begin{cases} p(c,\hat{r}) \times p(\hat{r},f) & \text{if } \hat{r} \text{ defined} \\ \max\{p(c,r) \times p(r,f) : r \in R\} & \text{otherwise} \end{cases} \tag{6.4c}$$

If we have not already selected a best relation, $\hat{r}$, then we define it as the relation, r, which corresponds to this pairwise combination score. Again, if there is more than one relation with the same score, then we choose the least common.

Step 3

Our final step attempts to assign relations to those concept/feature pairs which lack an exact mutually linking relation. This occurs around 17% of the time and is usually due to both the concept and feature terms being relatively low frequency.

We note that only a small proportion of the triples derive their relations in this way; at this point, in our training sets we had assigned relations to over 94% of the Wikipedia corpus, and 97% of the UKWAC corpus.

To solve this problem, we back off to semantic feature clusters. In other words, for a given feature, we consider all relations paired with other features in that cluster. We perform this clustering across all elements of the final potential feature set. We increase the total number of clusters to 500 to ensure each cluster doesn't contain an excessive number of features: the more features there are, the more relations there are to consider which reduces the likelihood of choosing an appropriate one.

Formally, we define $f_\star$ as the cluster for feature f, and $F_\star$ as the set of all feature clusters, and define the Feature Cluster Score analogously to our Pairwise Combination Score:

$$p(c,r) = \sum_{f \in F_\star} \frac{\text{freq}(c,r,f_\star)}{\text{freq}(c) \times \text{freq}(r)} \tag{6.5a}$$

$$p(r,f_\star) = \sum_{c \in C} \frac{\text{freq}(c,r,f_\star)}{\text{freq}(r) \times \text{freq}(f_\star)} \tag{6.5b}$$

$$\text{FCS}(c,f_\star) = \begin{cases} p(c,\hat{r}) \times p(\hat{r},f_\star) & \text{if } \hat{r} \text{ defined} \\ \max\{p(c,r) \times p(r,f_\star) : r \in R\} & \text{otherwise} \end{cases} \tag{6.5c}$$

As before, if we have not already selected a best relation, $\hat{r}$, then we define it as the relation, r, which corresponds to this Feature Cluster Score.

6.2.4 Reweighting

In our system's fourth and final stage we use the metrics derived above to assign an overall score for each triple using a weighting of parameters; we use the training set to derive the optimal values for these parameters. Our hope is that by using more parameters than in our previous experiments, and using ones which also take properties of the chosen relation into account, we will be better equipped to emulate the McRae norms. Each triple will again be assigned a score, and in addition to SVM, PMI and LL values, we also introduce our relation-related scores, namely Distinct Relation Score (DRS), Exact Match Score (EMS), Pairwise Combination Score (PCS) and our Feature Cluster Score (FCS).

As before, we will normalise the various scores so that they all lie between 0 and 1.

Our relation selection stage will already have fixed a relation, $\hat{r}$, for each concept and feature. We may then calculate for each of the triples $t = (c, \hat{r}, f)$ the overall score for that triple as:

$$\begin{aligned} \text{score}(t) = \beta_{\text{PMI}} \cdot \text{PMI}(t) + \beta_{\text{LL}} \cdot \text{LL}(t) + \beta_{\text{SVM}} \cdot \text{SVM}(t) + \beta_{\text{CFF}} \cdot \text{CFF}(t) \\ + \beta_{\text{DRS}} \cdot \text{DRS}(t) + \beta_{\text{EMS}} \cdot \text{EMS}(t) + \beta_{\text{PCS}} \cdot \text{PCS}(t) + \beta_{\text{FCS}} \cdot \text{FCS}(t) \end{aligned} \tag{6.6}$$

Given the extreme difficulty in matching based on relations (as already mentioned in previous chapters, and made doubly hard by our introduction of prepositions into some of the relations), we will optimise the parameters for superior feature performance.

We note that as the scores are relative to one another, we are free to fix one of the variables, but this still leaves the search-space of possible values extremely large. Cycling through all possibilities is $\mathcal{O}(n^7)$ where n is the number of evenly-spaced step-values of β tested in the range $[0, 1]$. Even when only testing a small range of such step-values, this would be extremely time-consuming. Therefore we employ a stochastic process to search for best-possible values for the parameters.

To achieve this, we use a random-restart hill-climbing algorithm. The algorithm starts at a random point, assigning the various β values a random value between 0 and 1. We assess the F-score at this starting point and store it. We next make a small perturbation to the point by adding a random value δ to each individual β value. We then assess if this new point offers a better F-score to the stored F-score: if so, we repeat the process starting from the new point and storing the new F-score, if not, we repeat the process from the old point. We repeat these steps over 500 iterations, gradually reducing the size of the movements (δ values) as we proceed. The random δ

movements are in the range [-1/k, 1/k] where $k = 10$ for the first 200 iterations. After the 200th iteration we assign $k = 20$; after the 300th, $k = 40$; and, after the 400th, $k = 60$.

We repeat this entire algorithm 1000 times, and choose the output (and β values) offering the best F-score across these 1000 attempts. The process is stochastic because it is non-deterministic: we use random variables to initialise the search, and repeating the experiment would likely produce different parameter-values even though the resulting F-scores would probably be similar. This repetition mitigates the issues associated with plateaux in such hill-climbing algorithms (where locally optimal but globally sub-optimal solutions are found). Given we will be applying these values to an unseen test set in our evaluation, we believe finding a globally optimal solution is not absolutely essential and that this process offers a reasonably good approximation of the best possible F-scores our system can produce and their corresponding β values.

The best values for the training parameters across the three corpora and their corresponding precision, recall and F-scores can be found in Tables 6.3 and 6.4 respectively.

Corpus	β_{PMI}	β_{LL}	β_{SVM}	β_{CFF}	β_{DRS}	β_{EMS}	β_{PCS}	β_{FCS}
Wikipedia	0.0065	0.0407	1.0000	0.0105	0.0293	0.0043	0.0424	0.0688
UKWAC	0.0001	0.0480	1.0000	0.0070	0.0004	0.0393	0.0103	0.0609
Combined	0.0010	0.0550	1.0000	0.0056	0.0148	0.0202	0.0250	0.0507

Table 6.3: Parameter estimation for Equation 6.6 across the two corpora and the combined corpus.

Corpus	Prec.	Recall	F
Wikipedia	0.2062	0.4165	0.2739
UKWAC	0.2089	0.4275	0.2803
Combined	0.2233	0.4567	0.2996

Table 6.4: Our best precision, recall and F-scores against the training (non-ESSLLI) norms when evaluating on features only, found using the β parameters highlighted in Table 6.3.

6.3 Evaluation

As in the previous experiment, we evaluate our system using a gold standard evaluation, human evaluation and our two semantic similarity evaluations.

To evaluate our system we again train it on the 466 non-ESSLLI concepts, and test on the ESSLLI 44. We again trained our system on both of the corpora individually as well as in combination.

Due to memory constraints during the machine learning stage associated with the very large number of training instances, we were only able to train the UKWAC models on one third of the UKWAC corpus; to retain the distribution of training instances in the corpus we selected every third learning pattern for training. The combined corpus was thus made up of the entirety of Wikipedia concatenated with this third of the UKWAC corpus.

6.3.1 Gold standard evaluation

We begin by comparing the output using the ESSLLI, synonym-expanded gold standard. The results can be found in Table 6.5.

Relation	Corpus	Prec.	Recall	F
With	Wikipedia	0.1131	0.2265	0.1509
	UKWAC	0.1000	0.2005	0.1335
	Combined	0.1214	0.2431	0.1620
With (aug.)	Wikipedia	0.1214	0.2431	0.1620
	UKWAC	0.1048	0.2101	0.1398
	Combined	0.1298	0.2598	0.1731
Without	Wikipedia	0.2798	0.5603	0.3732
	UKWAC	0.2560	0.5132	0.3416
	Combined	0.2798	0.5606	0.3733

Table 6.5: Our best precision, recall and F-scores against the synonym-expanded ESSLLI norms across the two corpora and the combined corpora set, found using the training parameters listed in Table 6.3. The augmented ('aug.') relation scores correspond to matching against 'synonym-expanded' relations, which also include the original relation text from the McRae norms.

It is perhaps unsurprising that our performance when including the relations in this evaluation is not as good as that of our previous experiment. We believe this is for two main reasons: 1) the relation set in the previous experiment was constrained by the relations extracted in the McRae norms, and our output only generated relations in this 'correct' format; 2) our new relation extraction allows for multi-word relations, something which the ESSLLI evaluation set does not accommodate in its current form. We could circumvent this issue by altering the evaluation methodology to ignore prepositional terms in the output, but this would render our efforts to extract them redundant. An alternative is to include the full text of the relations found in the original McRae norms into the expansion set as 'relation synonyms' for the lemmatised relations. We also include these augmented results in Table 6.5, under the 'With (aug.)' relation heading. Doing this means our final best F-score creeps up to 0.1731

for the combined corpus—this is our best 'with relation' ESSLLI score across all of our experiments.

We also note that performing these evaluations on the top ten properties returned further improves the situation (this is unsurprising since, as we have already mentioned, the ESSLLI set contains only ten properties per concept); for example, the precision on the combined corpus for the top ten evaluation of features only is 0.4409, and this result is despite our system being optimised for returning the best twenty features in the reweighting stage. Evaluating the top ten triples against the augmented relations returns a precision score of 0.2215 for the same corpus.

6.3.2 Human-generated semantic similarity comparison

Following the methodology described in Section 4.3.2, we again compare our system's output with human-generated semantic similarity scores. The results can be found in Table 6.6.

Comparing these results with the corresponding results from our previous two experiments (see Tables 4.5 and 5.6) we can see that the features only results are the best so far (with an average correlation of 0.75). What is also remarkable is that the evaluation with relations, with an average correlation of 0.63, exhibits only a minor drop from our semi-supervised learning experiment (average correlation 0.68)—in that experiment the relations had been derived directly from the training (non-ESSLLI) portion of the McRae norms, whereas in this experiment the relation extraction is completely unconstrained. We believe this to be an extremely encouraging result.

Relation	V	D	r	Conf. Int.
With	McRae	410	0.7853	[0.691, 0.854]
	Wikipedia	654	0.5977	[0.446, 0.716]
	UKWAC	712	0.6294	[0.486, 0.740]
	Combined	692	0.6714	[0.539, 0.771]
Without	McRae	355	0.7874	[0.693, 0.855]
	Wikipedia	478	0.7203	[0.603, 0.807]
	UKWAC	456	0.7543	[0.649, 0.832]
	Combined	475	0.7417	[0.632, 0.822]

Table 6.6: Pearson correlation (r) results and confidence intervals between the V_{Human} vectors and the similarity vectors V (and their vector dimensionalities D) from our best final experiment systems as reported in Table 6.5.

Relation	M	F	r	Conf. Int.
With	McRae	15.53	0.4721	[0.467, 0.477]
	Wikipedia	16.47	0.1156	[0.109, 0.122]
	UKWAC	16.60	0.1187	[0.112, 0.125]
	Combined	16.40	0.1155	[0.109, 0.122]
Without	McRae	15.32	0.4780	[0.473, 0.483]
	Wikipedia	15.05	0.1216	[0.115, 0.128]
	UKWAC	15.24	0.1563	[0.150, 0.163]
	Combined	15.09	0.1376	[0.131, 0.144]

Table 6.7: Frobenious distances, Pearson correlation (r) results and confidence intervals between the Leacock and Chodorow WordNet M_{LC} matrix and the similarity matrices M from our best final experiment systems as reported in Table 6.5.

		Judge				
Corpus		A	B	Avg	Kappa	Agreements
Wikipedia	c / p	202	204	203	0.6343	252 (84%)
	r / w	98	96	97		
UKWAC	c / p	193	204	198.5	0.7398	265 (88%)
	r / w	107	96	101.5		
Combined	c / p	212	216	214	0.7229	266 (89%)
	r / w	88	84	86		

Table 6.8: Inter-annotator agreement and judgements for our final extraction system applied to the three corpora.

6.3.3 WordNet semantic similarity comparison

We repeat the WordNet semantic similarity evaluation as described in Section 4.3.3 on our new output. The results can be found in Table 6.7. As in the previous semi-supervised learning experiment, our output does not correlate all that well with the WordNet similarity ratings, although the correlation is still positive. Although we do use WordNet cluster information as one of our machine learning attributes, its contribution in the final output is intentionally less than in our automatic extraction system. In any case, although the correlation is significantly less across all of the corpora than that found through the human semantic similarity ratings, we believe that stronger performance against human similarities is a more valuable result.

6.3.4 Human evaluation

Finally, we asked two native English speaking human judges to assess the accuracy of the output triples. As we were specifically aiming to hone the relation extraction abil-

	Judge			Judge	
knife	**A**	**B**	***pig***	**A**	**B**
sharpened by **hand**	c	c	*eat* **piglet**	c	p
based on **design**	c	c	*get* **fat**	c	c
made of **steel**	c	c	*produce* **pork**	r	c
be **small**	c	p	*breed* **farm**	r	r
pick on **fork**	r	r	*put into* **sausage**	c	c
be **make**	p	r	*be* **large**	p	p
crafted from **metal**	c	c	*have* **baby**	c	c
scaled for **use**	p	p	*be* **different**	p	p
make **cut**	c	c	*stunned through* **use**	r	w
be **sharp**	c	c	*be* **bacon**	c	r
be **weapon**	c	c	*be* **welfare**	r	r
have **edge**	c	c	*discover* **sheep**	c	c
have **handle**	c	c	*killed for* **meat**	c	c
be **serrated**	c	c	*used for* **food**	c	c
made of **stainless**	w	r	*label* **cattle**	w	w
is for **cutting**	c	c	*be* **animal**	c	c
have **blade**	c	c	*shackled by* **ham**	r	r
be **useful**	p	c	*chew* **tail**	c	c
be **tool**	c	c	*have* **disease**	c	c
be **dangerous**	c	c	*found in* **guinea**	c	c

Table 6.9: Our judges' assessments of the correctness of the top twenty relation/feature pairs for two concepts extracted from our final system, using the combined corpus.

ity of our system, we asked them to evaluate the full text of our extracted triples only (i.e., we did not ask them to evaluate the validity of just the concept and feature with no relation). The judges were unaware of the aims of the evaluation. We concatenate their ratings using the same methodology as for our previous human evaluations (Sections 4.3.4 and 5.3.4), however the instructions given to the participants were altered slightly to reflect that—as we now also have prepositional relations in the output—we no longer wished them to allow for absent prepositions. We include these instructions in Appendix B, Section B.3. The additional prepositional information (or lack thereof) in the relation terms could also improve the inter-annotator agreement scores. Therefore we again believe this evaluation offers an important insight into the viability of this method as a property extraction system, and indeed is arguably a stricter task than our previous human judgement evaluations. We report these results in Table 6.8, and show a sample of our output from the combined corpus and the corresponding judgements in Table 6.9.

It is clear that, as was already indicated by our gold standard, the best results are to

be found in the combined corpus, where an impressive 71.3% of the returned triples—including the relation—were marked as either plausible or correct with a Kappa score of 0.7229 indicating substantial agreement between the annotators. This constitutes an enormous improvement on our previous scores when including the relation (where just over half of the triples were judged as correct or plausible), which demonstrates the strong improvements derived from this novel relation extraction technique.

6.4 Discussion

This chapter has presented a strongly performing and fully automated system for unconstrained property norm extraction; the system employs both full parsing and chunking to extract features and relations respectively and introduces a novel multi-step backing-off method for relation selection. Our two human evaluations indicate that this is our best system overall to date, and furthermore its gold standard performance exceeds that of the current state of the art by a significant margin.

Potential criticisms of this system include the fact that although the three relation extraction steps together return relations for the majority of the concept/feature pairs, there is a small minority of relation-less triples. They are without relation for one of two reasons. The first is that no relations were extracted for the concept term during chunking. This is usually caused by concepts formed of two words are split by the chunker (e.g., ***sweet-potato***), thereby removing any possibility for a concept match in the second stage. The other possible cause for an *unknownrel* relation is singleton feature clusters, which prevent the semantic backing-off step from having any effect. Ways of solving these problems include splitting hyphenated/two-term concepts and checking for both terms in the NP chunk, or further backing-off to concept clusters. However as this problem only affected 0.03% of the generated triples we leave this for future work, which we discuss in the next chapter.

Chapter 7

Conclusions and future work

OUR FINAL CHAPTER DISCUSSES the primary contributions of our work, offers a number of potential avenues for future research into this highly challenging task and concludes with some final thoughts on the more theoretical implications of this research.

7.1 Contributions of our work

It is undeniable that our research aims are ambitious and our task challenging. We now outline what we believe to be the most important contributions of our research.

7.1.1 Extraction techniques

In total we developed four separate extraction systems for our task.

In the first experiment we created a first-attempt system using manually generated rules motivated by examining a subset of our concepts and their correct features in a small, targeted training corpus (our Wiki500 corpus). Here we developed a number of insights into how syntactic structure can indicate property norm-like information and how the domain of the corpus affects the output. We also investigated the utility of semantic WordNet-based clustering in sorting through potential features, and to this end we explored a number of different clustering techniques for this subtask. Our first system's performance was reasonable, and formed the baseline for subsequent experiments.

In the automatic extraction system, we aimed to improve our initial feature extraction by revising our rules to make them more restrictive in their initial search, investigating the viability of a pre-extraction step (finding potential features to be used as additional input to the rules), switching to a more accurate parser, and revising our

entire ruleset to take grammatical relation directionality into account when traversing POS-GR graphs. The second stage of this system introduced a number of potential reweighting factors (including a novel entropy measure), and we used the training set to optimise these parameters, aiming to render the output more property-norm like. We evaluated this and all of our subsequent experiments blind: training on one subset of the McRae norms and evaluating on the remainder.

This experiment also showed that the results when using two distinct 'types' of corpora were superior across nearly all of our evaluations compared to the results from each individual corpus. This phenomenon could be ascribed purely to the increased size of a combined corpus, however we believe—given the different nature of the triples extracted from each corpus and how these usually came together to form the combined output—it was as much a product of the diversity of corpora as it was of their size.

Having explored the various factors and identifying syntactic/lexical information which typically flagged a potential property norm-like relation, in our next experiment we approached our task as one of relation classification by training a support vector machine to automatically detect the GR-POS graph-derived attributes which were likely to be indicative of such relationships. This system worked under the assumption that all adjectives and nouns which co-occurred with a concept were potential features for that concept, and the SVM was therefore used to classify those relationships into relations found in the training set. This gave us both a relation and SVM score (which measured the confidence of the prediction) for each feature, which we again reweighted with other metrics to derive the final output. This method gave us reasonable results when considering the extracted triples including the relation, but performed particularly well when extracting salient features for concepts (F-score of 0.3225, and 76.8% correct or plausible features when evaluated by humans).

In our final experiment, we wished to take advantage of the promising results from our semi-supervised learning method—specifically, the strong performance in terms of feature extraction—yet also harness the large amounts of information which we ignored in that method (where we modelled our system only on sentences which matched up with the property norm training set). Therefore we took a multi-stage approach, wherein we would, as before, use support vector machines to acquire promising features from the corpus (using an even more comprehensive machine learning attribute set) and then use those features, together with their corresponding concepts, to anchor our search for plausible relations in the corpus. We introduced a novel backing-off method to find the most likely relation for a concept/feature pair, and introduced a more refined relation representation. Our backing-off method also produced a num-

ber of additional metrics which could act as potential indicators of true relations. We completed the training of this system by using a stochastic search algorithm to find the optimal reweighting of our metrics, old and new. The resulting system produced output which beat all of our previous systems on both the gold standard and the two human judgement-derived evaluations.

We believe that the evolution of these techniques has been extremely instructive—culminating in our final system which achieves state of the art performance on this task—and represents a significant contribution to this research domain.

7.1.2 Structure of property norm-like information in text

For our extraction systems to function properly it was essential that we understood how information contained in property norms was likely to appear in normal corpus text; the first two experiments offered explicit descriptions (in the form of our extraction rules) of the underlying linguistic structures likely to indicate such properties. These initial insights allowed us to provide our semi-supervised learning systems with appropriate machine learning attributes, and generalise across them to better find relationships between concepts and features. The final experiment demonstrated that while parsing can aid in finding relevant features for concepts, the lexical form of phrases can prove extremely useful in making concrete the exact nature of the relationship between concept and feature, without recourse to a limited set of training relations. In sum, we believe that we have gained a much deeper understanding of the types and patterns of linguistic structure which are likely to indicate conceptual properties for concepts.

7.1.3 Evaluation methodologies

As our research has demonstrated, finding an accurate and reliable evaluation methodology remains a serious obstacle in assessing the performance of any system tackling this task.

Our main gold standard was derived from the McRae norms in the form of the ESSLLI evaluation subset. As already mentioned, the work of Baroni et al. (2009) is relevant to our own. Their approach achieved a precision score of 0.239 on the top ten returned features evaluated against the ESSLLI gold standard: our final system offered a precision of 0.4409 on the same evaluation. Moreover, Baroni et al. did not explicitly derive relation terms; when we include the extracted relation terms in the same evaluation we achieve precision of 0.2216, almost matching Baroni et al.'s features-only score.

However, as we have discussed at length (and demonstrated by way of other evaluations), we believe this gold standard is not sufficient for a comprehensive and fair evaluation of this task. Therefore we began by exploring whether—given the ultimate aims of the research within cognitive science—using techniques based on EEG and fMRI data derived from activity in the human brain could prove fruitful. Unfortunately the results appeared to indicate that such evaluation methodologies may yet be too ambitious for a task such as this. We are somewhat agnostic as to how this could best be remedied, be it by further advances in brain activity monitoring technology, a more enhanced understanding of how the brain conceptualises objects in the world or by improvement in the quality of the system's output; the answer is most likely a combination of the three.

We also explored a conceptual structure statistics evaluation; these results showed a majority of correlations between our pilot output and the McRae norms, however our output did not exhibit the structural differences between distinct categories of concepts (living and non-living) which we might have expected. Due to the way in which it evaluates output in aggregate, we believe this evaluation methodology could be appropriate as a test of the overall human-like nature of a property norm extraction system.

In order to circumvent these problems in another way, we introduced a novel property evaluation method based on similarity matrices. This method has shown the various systems to be capable of producing binary-concept similarity which correlates with Leacock and Chodorow's WordNet similarity metric, although the positive correlation did decline when using systems which did not overtly rely on WordNet for their final output. We also evaluated our systems' capacity to predict human ratings of semantic similarity; our final extraction system exhibited extremely strong performance, showing an ability to predict human-rated semantic similarity on par with that of the McRae norms themselves.

Finally, we have employed direct human annotation to comprehensively evaluate our systems; an extremely labour-intensive evaluation technique, but one that remains absolutely necessary for a challenging task such as this. This last point is illustrated by the notable discrepancies across all of our experiments between the gold standard evaluation and human evaluation, where a great number of 'incorrect' triples (according to the ESSLLI standard) were in fact deemed to be correct or plausible by human annotators.

7.2 Future work

Over the course of this research we have touched on a range of issues, tasks and techniques spanning the field of Natural Language Processing and there are many potential new directions for this research to take. We list those which we think will lead to the strongest gains in performance.

7.2.1 Corpora

Our work has already shown the benefits that may be derived from combining multiple types of corpora: a simple concatenation of extracted triples from the two corpora offered an immediate improvement across the board. NLP techniques tend to perform better with more data, therefore future work could employ further, larger or more task-appropriate corpora. For example, one could employ a corpus of children's literature (e.g., Sealey and Thompson (2004) used a subset of the BNC containing only texts written for children) or a 'basic English' corpus (e.g., Ruiz-Casado et al. (2005) used Simple English Wikipedia to automatically extract semantic relations for WordNet). Our system tends to perform better with shorter sentences because the grammatical relation paths are shorter and consequently less error-prone, and the type of common sense data often found in such corpora (e.g., statements of fact obvious to adult readers but not to learners) could be of enormous benefit. We could also consider using an empirically derived, more sophisticated weighting of corpora to maximise accuracy—for example, certain corpora may be better for certain types of relations/features.

Another option would be to follow the example of Etzioni et al. (2011) and employ a web-scale corpus. This could prove useful for very low frequency concepts and features. For example, in our final experiment a web-scale corpus could have been very helpful in finding directly linking relations between concepts and features and would further reduce the need to back off during the Relation Selection stage.

7.2.2 Property representation

Improving the property representation is another potential avenue of investigation. In our first two experiments, we extracted three-word triples in a ***concept*** *relation* **feature** structure, where only one word was allowed per field. Future, more sophisticated representations could harness the flexibility in feature-derivation offered by our two path-based rule construction systems (i.e., the fact we can extract more than one node from within a matched path). Our semi-supervised learning method allowed for multi-word relations, however these were constrained by what already appeared in

the McRae norms. Only in our final experiment did we allow for more flexibility in the relation field, and indeed the human evaluation reflected how important this was (our inter-annotator agreements reached their highest levels with the additional information). But in general, our representation structure sometimes made it impossible to preserve some discriminating information as it appeared in the norms and in corpora. For example, a property of ***giraffe*** is *has* **long neck**—in all the experiments, this would have been reduced to ***giraffe*** *has* **neck**, which clearly doesn't encapsulate the property's distinctive nature. According to current cognitive psychology theories these distinctive properties are an essential component of the brain's representation of concepts.

Therefore it might be worthwhile investigating the possibility of further enhancing the property representation beyond its current form to a more flexible representation. For example, we could use a representation framework specific to our task with multiple (possibly empty) slots for the different components of a prototypical property:

(***concept***, <*verb*>, <preposition>, <feature-modifier>, <**feature**>)

so the properties *duck swims*, *giraffe has long neck* and *pan used for cooking* would have the representations:

(*duck, swims*, **null**, **null**, **null**)
(*giraffe, has*, **null**, *long, neck*)
(*pan, used, for*, **null**, *cooking*)

But even that does not quite encapsulate all possibilities; for example the behavioural properties (***lion*** – **roars**) do not fit neatly into such a structure without certain, possibly undesirable, trade-offs (e.g., transformation into the somewhat unnatural triple ***lion*** *do* **roar**). Finally, a number of training concepts, e.g., ***rocking horse*** and ***sailing boat***, were formed of two terms, and all our systems struggled to deal with such compound nouns. We have touched on a number of the issues associated with conceptual property representation but there are clearly many areas still to explore.

7.2.3 Word sense disambiguation

Another possible research direction would be the implementation of differentiation between words with multiple meanings to ensure the 'correct' properties are returned. This is exemplified by our system's output for a number of polysemous concepts, such as ***bat*** and ***fan***, where there are properties generated which are appropriate to only one of the concept's meanings (e.g., our final system returns both *have* **wings** and *hit* **ball** as properties of ***bat***). Indeed, McRae's human annotators were given disambiguating information for such concepts—they were asked to rate ***bat (baseball)*** or ***bat (animal)***, and

fan was phrased as ***fan (appliance)***. One could therefore imagine a system which takes these disambiguating terms into account. One could, for example, use topic classification techniques to draw only from parts of the corpus which are likely to be directly relevant to the desired concept. Indeed, this technique could also be applicable to other concepts which aren't ambiguous in a strict sense but where the interpretation of the word is highly dependent on context. For example, the concept ***bag*** often appears in a compound noun form—***shopping bag***, ***bin bag***, ***sleeping bag***—and the combination of extractions from these could distort the semantically distinct properties which we seek. However, we note that current word sense disambiguation techniques are not totally accurate, and one therefore ought to be wary of excluding potentially accurate properties when considering subtly polysemous concepts.

7.2.4 Making the output more property norm-like

We could also investigate further reweighting factors likely to yield human-like norms: for example the *t*-test of word association by Manning and Schütze (1999). A potential criticism of our reweighting method is that the various metrics we use (our SVM score, entropy, PMI and log-likelihood statistics, and, in our final experiment, a range of novel measures) do not necessarily scale linearly, so it might not make sense to reweight them as if they did. One could consider replacing our parameter optimisation techniques with the application of another support vector machine. However this would probably require a lot more training data of correct/incorrect triples to make a significant difference—we discuss options for this next.

7.2.5 Collecting training data

One of the major difficulties we encountered when developing the various systems was that of a lack of sufficient training data; the norms we had access to were incomplete insofar as there were a large number of properties which were marked as correct and true by human evaluators but which did not appear in the norms.

We would therefore consider setting up a publicly accessible web-based system to enable large-scale and rapid evaluation of our system's output. This would enable us to quickly obtain large amounts of human-generated feedback in a consistent and rigorous fashion, thereby circumventing the major issues associated with evaluation when using a static 'gold standard' for cognitive activation patterns. This could prove especially helpful in providing further training data for our semi-supervised learning stages. We could also make data collection dynamic. For example, one could use active-learning to introduce a feedback loop of human annotation to better distinguish

between relations and features which the algorithm tends to misclassify, and supplement input pattern data with disambiguating information to draw a distinction between valid and non-valid relations. In our systems this was achieved only implicitly by those terms' absence in the gold standard, however feedback which strongly indicated to our learner that properties such as *be* **many** were uninteresting could prove invaluable in getting closer to a conceptual structure-like representation.

One possible strategy for such large-scale data collection would be through the use of crowd-sourced labour (e.g., Amazon Mechanical Turk). An alternative would be to create a system similar to reCAPTCHA (Von Ahn et al., 2008): when wishing to verify that a website user was indeed human, one could present users with a set of property norm-like statements, some known (i.e., true statements from the norms or false statements from previous human evaluations) and some unknown (i.e., from the system's output), and ask the user to evaluate them. A sufficient number of identical responses for the unknown statements from human-verified users would establish the statements as correct or not.

7.3 Final thoughts

We conclude by discussing the theoretical implications of our research. We first draw attention to the fact that techniques for extracting properties of concepts—rather than mere word associations or concept-similarity predictors—are still in their infancy. Furthermore, they must be viewed as distinct and, we hope, more useful than semantic classifiers/cluster generators, at least in the context of cognitive psychology.

We acknowledge that the properties we extract are not 'behavioural' in the same way that McRae's norms are; rather they are, by their derivation, a product of our chosen rules and later by their statistical distribution in text. We are not contending that the properties are exactly equivalent to those found in people's brains or its equivalent representation (i.e., conceptual knowledge). Rather we are aiming to demonstrate the breadth and richness of semantic information that it is possible to extract from large bodies of text, as well as the potential to generate a high proportion of the conceptual properties which humans know, with the potential added benefit of being able to use these properties for research in cognitive psychology.

An important criticism of property norming studies, from a cognitive psychology perspective, is that although they are interesting inasmuch as the presence/absence of certain properties is telling in itself, they are incomplete in terms of building a full and comprehensive property-based description of a certain concept: the fact that humans could most likely surmise the identity of a concept given only its properties from, for

example, the McRae norms is a product of the participants' pre-existing conceptual knowledge which 'fills in the gaps'. In our view, an ideal system would generate all properties for a given concept, ranking them in terms of their specificity and salience for the concept at hand. This work aims to fill in more of these 'gaps', but it is as yet unknown whether we can totally emulate all conceptual knowledge about a given concept. What is, however, clear is that a significant amount of semantic information—as rich as that found in property norms—can be gleaned purely from data in language. This may seem like a manifest statement, but it is an important one nonetheless. Our system's output is a product of the functional structure of language, rather than the structure of human knowledge, or the world itself—this distinction is key if only because we still don't fully the understand nature of the differences/similarities between these categories. Yet it also raises the question of whether or not, given appropriate NLP techniques and a sufficiently large corpus (which would effectively be the product of many humans' thoughts and statements, derived both from their linguistic interactions and their experience of the world itself), it would be possible to comprehensively emulate the conceptual representation of a given concept by a layman. Can knowledge in language (particularly with the advent of the web providing the sum of many people's linguistic output) adequately mirror that in the brain or the world?

We leave it as an open question which technique cognitive psychologists can or should employ in attempting to understand how language is conceptualised in the brain. This could be the sometimes changing, often idiosyncratic (perhaps language/culture dependent) and highly concept-specific output of humans citing concepts for a specific rule; or a technique which is less intuitive by human standards but arguably offers a more scientific, consistent and uniform approach to the representation of concepts.

The development and implementation of accurate property extraction methods and their evaluation is an exciting, challenging, and relatively new task. This work demonstrates that there is a wealth of semantic knowledge to be gained from language itself, and that reasonable results in extracting it are possible through the combination of a number of NLP techniques.

Bibliography

A. Almuhareb and M. Poesio. 2004. Attribute-based and value-based clustering: An evaluation. In *Proceedings of EMNLP 2004*, pages 158–165. Association for Computational Linguistics.

A. Almuhareb and M. Poesio. 2005. Concept learning and categorization from the web. In *Proceedings of CogSci*.

M. Andrews and G. Vigliocco. 2010. The Hidden Markov Topic Model: A Probabilistic Model of Semantic Representation. *Topics in Cognitive Science*, 2(1):101–113.

M. Andrews, G. Vigliocco, and D. Vinson. 2009. Integrating experiential and distributional data to learn semantic representations. *Psychological Review*, 116(3):463.

M.H. Ashcraft. 1978. Property norms for typical and atypical items from 17 categories: A description and discussion. *Memory & Cognition*, 6(3):227–232.

J. Baldridge. 2005. The Apache OpenNLP project.

E. Barbu. 2008. Combining methods to learn feature-norm-like concept descriptions. In *Proceedings of the ESSLLI Workshop on Distributional Lexical Semantics*, pages 9–16.

M. Baroni and A. Lenci. 2008. Concepts and properties in word spaces. *Italian Journal of Linguistics*, 20(1):55–88.

M. Baroni and A. Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

M. Baroni, S. Bernardini, F. Comastri, L. Piccioni, A. Volpi, G. Aston, and M. Mazzoleni. 2004. Introducing the La Repubblica corpus: A large, annotated, TEI (XML)-compliant corpus of newspaper Italian. In *In LREC 2004*, pages 1771–1774.

M. Baroni, S. Evert, and A. Lenci, editors. 2008. *ESSLLI 2008 Workshop on Distributional Lexical Semantics*.

M. Baroni, B. Murphy, Barbu E., and Poesio M. 2009. Strudel: A corpus-based semantic model based on properties and types. *Cognitive Science*, pages 1–33.

S. Bird. 2006. NLTK: The natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

D.M. Blei, A.Y. Ng, and M.I. Jordan. 2003. Latent Dirichlet Allocation. *The Journal of Machine Learning Research*, 3:993–1022.

T. Briscoe. 2006. An introduction to tag sequence grammars and the RASP system parser. *Computer Laboratory Technical Report*, 662.

A. Budanitsky and G. Hirst. 2006. Evaluating Wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

P. Buitelaar, P. Cimiano, and B. Magnini. 2005. Ontology learning from text: methods, evaluation and applications. *Computational Linguistics*, 32(4).

A. Caramazza and J.R. Shelton. 1998. Domain-specific knowledge systems in the brain: The animate-inanimate distinction. *Journal of Cognitive Neuroscience*, 10(1):1–34.

J. Carroll and T. Briscoe. 2002. High precision extraction of grammatical relations. In *Proceedings of the 19th International Conference on Computational Linguistics–Volume 1*, page 7. Association for Computational Linguistics.

K.W. Church and P. Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

S. Clark and J.R. Curran. 2007a. Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics*, 33(4):493–552, 1.

S. Clark and J.R. Curran. 2007b. Formalism-independent parser evaluation with CCG and DepBank. In *Annual Meeting–Association for Computational Linguists*, volume 45, page 248, 12.

S. Clark and D. Weir. 2002. Class-based probability estimation using a semantic hierarchy. *Computational Linguistics*, 28(2):187–206.

J. Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*.

M. Collins and Y. Singer. 1999. Unsupervised models for named entity classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 189–196.

C. Cortes and V. Vapnik. 1995. Support-vector networks. *Machine Learning*, 20(3):273–297.

K. Crammer and Y. Singer. 2002. On the algorithmic implementation of multiclass kernel-based vector machines. *The Journal of Machine Learning Research*, 2:265–292.

G.S. Cree, C. McNorgan, and K. McRae. 2006. Distinctive features hold a privileged status in the computation of word meaning: Implications for theories of semantic memory. *Journal of Experimental Psychology Learning Memory and Cognition*, 32(4):643.

J. Curran and S. Clark. 2003. Investigating GIS and smoothing for maximum entropy taggers. In *Proceedings of the 10th conference of the European chapter of the Association for Computational Linguistics*, pages 91–98. Association for Computational Linguistics.

D. Davidov, A. Rappoport, and M. Koppel. 2007. Fully unsupervised discovery of concept-specific relationships by web mining. In *Annual Meeting–Association For Computational Linguistics*, volume 45, page 232.

S.C. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R.A. Harshman. 1990. Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science*, 41:391–407.

B. Devereux, N. Pilkington, T. Poibeau, and A. Korhonen. 2009. Towards unrestricted, large-scale acquisition of feature-based conceptual representations from corpus data. *Research on Language & Computation*, pages 1–34.

B. Devereux, C. Kelly, and A. Korhonen. 2010. Using fMRI activation to conceptual stimuli to evaluate methods for extracting conceptual representations from corpora. In *First Workshop on Computational Neurolinguistics*, page 70.

J.T. Devlin, L.M. Gonnerman, E.S. Andersen, and M.S. Seidenberg. 1998. Category-specific semantic deficits in focal and widespread brain damage: A computational account. *Journal of Cognitive Neuroscience*, 10(1):77–94.

T. Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

O. Etzioni, M. Cafarella, D. Downey, A.M. Popescu, T. Shaked, S. Soderland, D.S. Weld, and A. Yates. 2005. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence*, 165(1):91–134.

O. Etzioni, A. Fader, J. Christensen, S. Soderland, and M.T. Center. 2011. Open Information Extraction: The Second Generation. In *Twenty-Second International Joint Conference on Artificial Intelligence*.

M.J. Farah and J.L. McClelland. 1991. A computational model of semantic memory impairment: Modality specificity and emergent category specificity. *Journal of Experimental Psychology: General*, 120(4):339–357.

C. Fellbaum. 1998. *WordNet: An electronic lexical database*. The MIT press.

A. Ferraresi, E. Zanchetta, M. Baroni, and S. Bernardini. 2008. Introducing and evaluating UKWAC, a very large web-derived corpus of English. In *Proceedings of the 4th Web as Corpus Workshop (WAC-4) – Can we beat Google?*, pages 47–54.

R.A. Fisher. 1915. Frequency distribution of the values of the correlation coefficient in samples from an indefinitely large population. *Biometrika*, 10(4):507–521.

J.L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

P. Garrard, M.A.L. Ralph, J.R. Hodges, and K. Patterson. 2001. Prototypicality, distinctiveness, and intercorrelation: Analyses of the semantic attributes of living and nonliving concepts. *Cognitive Neuropsychology*, 18(2):125–174.

J. Giménez and L. Marquez. 2004. SVMTool: A general POS tagger generator based on support vector machines. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*.

S. Godbole, S. Sarawagi, and S. Chakrabarti. 2002. Scaling multi-class support vector machines using inter-class confusion. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 513–518.

R. Grondin, S.J. Lupker, and K. McRae. 2009. Shared features dominate semantic richness effects for concrete concepts. *Journal of Memory and Language*, 60(1):1–19.

M.A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th Conference on Computational Linguistics*, volume 2, pages 539–545. Association for Computational Linguistics.

M.A. Hearst. 1998. Automated discovery of WordNet relations. *WordNet: an electronic lexical database*, pages 131–151.

S.A. Huettel, A.W. Song, and G. McCarthy. 2009. *Functional Magnetic Resonance Imaging*. Sinauer Associates.

J.J. Jiang and D.W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of International Conference Research on Computational Linguistics (ROCLING X)*.

T. Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. *Machine Learning: ECML-98*, pages 137–142.

T. Joachims. 1999. Making large scale SVM learning practical.

D. Jurafsky and J.H. Martin. 2000. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, volume 2. Prentice Hall.

J. Kazama, T. Makino, Y. Ohta, and J. Tsujii. 2002. Tuning support vector machines for biomedical named entity recognition. In *Proceedings of the ACL-02 Workshop on Natural Language Processing in the Biomedical Domain*, volume 3, pages 1–8. Association for Computational Linguistics.

C. Kelly, B. Devereux, and A. Korhonen. 2010. Acquiring human-like feature-based conceptual representations from corpora. In *First Workshop on Computational Neurolinguistics*, page 61. Association for Computational Linguistics.

C. Kelly, B. Devereux, and A. Korhonen. 2012. Semi-supervised learning for automatic conceptual property extraction. In *Proceedings of the 3rd Workshop on Cognitive Modeling and Computational Linguistics*, pages 11–20. Association for Computational Linguistics.

C. Kelly, A. Korhonen, and B. Devereux. 2013. Minimally supervised learning for unconstrained conceptual property extraction. In *Proceedings of the 35th Annual Conference of the Cognitive Science Society*.

C. Kelly, B. Devereux, and A. Korhonen. to appear. Automatic extraction of property norm-like data from large text corpora. *Cognitive Science*.

J.R. Landis and G.G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

C. Leacock and M. Chodorow. 1998. Combining local context and WordNet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283.

G. Leech. 1992. 100 million words of English: the British National Corpus. *Language Research*, 28(1):1–13.

D.B. Lenat. 1995. CYC: A large-scale investment in knowledge infrastructure.

D. Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*, pages 296–304.

C.J. Lin. 2007. Projected gradient methods for non-negative matrix factorization. *Neural Computation*, 19(10):2756–2779.

K. Lund and C. Burgess. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. In *Behaviour Research Methods, Instruments, and Computers, 28*, pages 203–208.

A. Maedche and S. Staab. 2001. Ontology learning for the semantic web. *Intelligent Systems, IEEE*, 16(2):72–79.

C.D. Manning and H. Schütze. 1999. *Foundations of Statistical Natural Language Processing*, volume 59. MIT Press.

M.E.J. Masson. 1995. A distributed memory model of semantic priming. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 21(1):3.

K. McRae, V.R. De Sa, and M.S. Seidenberg. 1997. On the nature and scope of featural representations of word meaning. *Journal of Experimental Psychology-General*, 126(2):99–130.

K. McRae, G.S. Cree, M.S. Seidenberg, and C. McNorgan. 2005. Semantic feature production norms for a large set of living and nonliving things. *Behavioral Research Methods, Instruments, and Computers*, 37:547–559.

K. McRae. 2012. Personal communication with Ken McRae.

M. Mintz, S. Bills, R. Snow, and D. Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, volume 2, pages 1003–1011. Association for Computational Linguistics.

T.M. Mitchell, S.V. Shinkareva, A. Carlson, K.M. Chang, V.L. Malave, R.A. Mason, and M.A. Just. 2008. Predicting human brain activity associated with the meanings of nouns. *Science*, 320(5880):1191.

H.E. Moss, L.K. Tyler, and J.T. Devlin. 2002. The emergence of category-specific deficits in a distributed semantic system. *Category specificity in brain and mind*, pages 115–148.

H.E. Moss, L.K. Tyler, and K.I. Taylor. 2007. Conceptual structure. *The Oxford Handbook of Psycholinguistics*, pages 217–234.

B. Murphy, M. Baroni, and M. Poesio. 2009. EEG responds to conceptual stimuli and corpus semantics. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 619–627. Association for Computational Linguistics.

G. Murphy. 2002. *The Big Book of Concepts*. The MIT Press.

B. Omelayenko. 2001. Learning of ontologies for the web: the analysis of existent approaches. In *Proceedings of the International Workshop on Web Dynamics*.

S. Padó and M. Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

P. Pantel and D. Lin. 2002. Discovering word senses from text. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 613–619.

P. Pantel and M. Pennacchiotti. 2008. Automatically harvesting and ontologizing semantic relations. In *Proceedings of the 2008 Conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge*, pages 171–195. IOS Press.

K. Pearson. 1901. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

P.M. Pexman, S.J. Lupker, and Y. Hino. 2002. The impact of feedback semantics in visual word recognition: Number-of-features effects in lexical decision and naming tasks. *Psychonomic Bulletin and Review*, 9(3):542–549.

H. Poon and P. Domingos. 2010. Unsupervised ontology induction from text. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 296–305. Association for Computational Linguistics.

B. Randall, H.E. Moss, J.M. Rodd, M. Greer, and L.K. Tyler. 2004. Distinctiveness and correlation in conceptual structure: Behavioral and computational studies. *Journal of Experimental Psychology Learning Memory and Cognition*, 30(2):393–406.

A. Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, volume 1, pages 133–142.

P. Resnik. 1995. Using information content to evaluate semantic similarity. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 448–453.

T.C. Rindflesch, L. Tanabe, J.N. Weinstein, and L. Hunter. 2000. EDGAR: extraction of drugs, genes and relations from the biomedical literature. In *Pacific Symposium on Biocomputing*, page 517.

E. Rosch and C.B. Mervis. 1975. Family resemblances: Studies in the internal structure of categories. *Cognitive Psychology*, 7(4):573–605.

M. Ruiz-Casado, E. Alfonseca, and P. Castells. 2005. Automatic extraction of semantic relationships for WordNet by means of pattern learning from Wikipedia. *Natural Language Processing and Information Systems*, pages 67–79.

A. Sealey and P. Thompson. 2004. What do you call the dull words? Primary school children using corpus-based approaches to learn about language. *English in Education*, 38(1):80–91.

P. Singh, T. Lin, E.T. Mueller, G. Lim, T. Perkins, and W. Li Zhu. 2002. Open Mind Common Sense: Knowledge acquisition from the general public. *Lecture Notes in Computer Science*, pages 1223–1237.

M. Steyvers. 2010. Combining feature norms and text data with topic models. *Acta Psychologica*, 133(3):234–243.

M.J. Sussna. 1994. Verb semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting for the Association for Computational Linguistics*, pages 133–138.

K.I. Taylor, A. Salamoura, B. Randall, H. Moss, and L.K. Tyler. 2008. Clarifying the nature of the distinctiveness by domain interaction in conceptual structure: Comment on Cree, McNorgan, and McRae (2006). *Journal of Experimental Psychology Learning Memory and Cognition*, 34(3):719.

K.I. Taylor, B. Devereux, K. Acres, B. Randall, and L.K. Tyler. 2011. Contrasting effects of feature-based statistics on the categorisation and basic-level identification of visual objects. *Cognition*, 122(3):363–74.

I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. 2004. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the 21st International Conference on Machine Learning*, page 104.

P.D. Turney. 2001. Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of the 12th European Conference on Machine Learning*, EMCL '01, pages 491–502. Springer-Verlag.

L.K. Tyler, H.E. Moss, M.R. Durrant-Peatfield, and J.P. Levy. 2000. Conceptual structure and the structure of concepts: A distributed account of category-specific deficits. *Brain and Language*, 75(2):195–231.

A. Vinokourov, J. Shawe-Taylor, and N. Cristianini. 2003. Inferring a semantic representation of text via cross-language correlation analysis. *Advances in Neural Information Processing Systems*, 15:1473–1480.

D.P. Vinson and G. Vigliocco. 2008. Semantic feature production norms for a large set of objects and events. *Behavior Research Methods*, 40(1):183.

L. Von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum. 2008. ReCAPTCHA: Human-based character recognition via web security measures. *Science*, 321(5895):1465–1468.

F. Wu and D.S. Weld. 2010. Open Information Extraction using Wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 118–127. Association for Computational Linguistics.

W. Xu, X. Liu, and Y. Gong. 2003. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 267–273.

Appendix A

Semantic similarity instructions

A.1 Initial instructions

You will be presented with pairs of words that refer to well-known concepts in the world (e.g., `turtle <--> kettle`, `lion <--> dog`). Your task is to rate, on a scale of 1 to 7, how similar in meaning the two concepts are.

You will be asked to rate 90 concept-concept pairs in total. The entire set should take you no more than 20 minutes to complete. Try to use the entire ratings scale when making your responses.

A.2 Instructions for each concept-concept pair

Please rate how similar the following concrete nouns are on a scale of 1 to 7, 1 meaning 'very dissimilar'; and 7 meaning 'very similar'.

Appendix B

Human triple evaluation instructions

B.1 Triple evaluation

In this experiment, you are asked to judge whether properties listed for concepts are true or not. The properties are listed as word triples of the form `<concept> <relation> <feature>`, where `<concept>` is a noun (e.g., 'tiger'), `<feature>` is a noun or adjective (e.g., 'stripe') and `<relation>` is a verb representing the link between them (e.g., 'have').

Some examples of valid triples are listed below

```
tiger be animal
tiger live jungle
accordion produce music
accordion require air
```

Note that prepositions are not included in relations. So

```
tiger live jungle
accordion wear chest
airplane find airport
tiger use circus
```

would be true features, because tigers live in jungles, accordions are worn on chests, airplanes are found at airports, and tigers are used by circuses. You may assume absent prepositions when making your judgments.

Features need not be true of all instances of the concepts. So `tiger use circus` and `airplane do crash` are correct features, even though not all tigers are used by circuses and not all airplanes crash.

All feature terms have been made singular, so `tiger have tooth` and `accordion have key` would be correct triples, even though tigers have more than one tooth and accordions have more than one key.

Note that sometimes the concept noun is the agent of the relation verb, and sometimes the feature word is. So, for example, `airplane use passenger` is a valid feature of airplane, because passengers use airplanes.

Some features that you will see will be true, and some will be untrue. When judging the correctness of each `<concept> <relation> <feature>` triple, we would like you to select between four possibilities:

c correct. Triple represents a correct, valid feature (e.g., `tiger be animal`, `airplane use passenger`).

p: plausible. Triple is not correct, but the triple may be plausible in a very specific set of circumstances (e.g., `tiger exhibit dimorphism`, `airplane land movie`) and/or the triple may be very general (e.g., `airplane be available`, `airplane have version`), or may be partly correct (e.g., `tiger be black`).

r: wrong, but related. The triple is wrong, but there is some kind of relationship between concept and the relation and/or feature (e.g., `motorcycle be car`, `accordion sing polka`, `accordion play astronaut`).

w: just completely wrong (e.g., `accordion fall Mississippi`, `tiger debunk reputation`).

Please use your own subjective judgment when making your decisions.

B.2 Concept/feature evaluation

In this part of the experiment, you are asked to judge whether semantic features—without relations—listed for concepts are true or not. The features are listed as word pairs of the form `<concept> --> <feature>`, where `<concept>` is a target concept noun (e.g., 'tiger') and `<feature>` is a noun or adjective (e.g., 'stripe'). Your task is to decide whether there is a relationship between the two words, and the strength of that relationship.

Some examples of valid pairs are listed below

```
tiger --> animal
tiger --> jungle
accordion --> music
accordion --> air
```

You may assume any correct/plausible relationship when making your judgments.

Note that features need not be true of all instances of the concepts. So `tiger --> circus` and `airplane --> crash` are correct features, even though not all tigers are used by circuses and not all airplanes crash.

Some pairs that you will see will represent true relationships, and some will be untrue. When judging the correctness of each `<concept> --> <feature>` pair, we would like you to select between four possibilities:

c: correct. Pair represents a correct, valid feature (e.g., `tiger --> animal`, `airplane --> passenger`).

p: plausible. Pair is not correct, but the pair may be plausible in a very specific set of circumstances (e.g., `tiger --> dimorphism`, `airplane --> terrorist`) and/or the pair may be very general (e.g., `airplane --> available`, `airplane --> large`), or may be partly correct (e.g., `tiger --> black`).

r: wrong, but related. The pair is wrong (i.e., not directly related), but there is some kind of tangential relationship between concept and the feature (e.g., `motorcycle --> screwdriver`, `airplane --> spaceship`).

w: just completely wrong (e.g., `airplane --> daisy`, `tiger --> lightbulb`).

Please use your own subjective judgment when making your decisions.

B.3 Triple evaluation with prepositions

In this experiment, you are asked to judge whether properties listed for concepts are true or not. The properties are listed as word triples of the form `<concept> <relation> <feature>`, where `<concept>` is a noun (e.g., 'tiger'), `<feature>` is a noun, adjective or verb (e.g., 'stripe', 'ferocious', 'roar', 'jungle') and `<relation>` represents the link between them (e.g., 'have', 'be', 'do', 'lives in').

Some examples of valid triples are listed below

```
tiger be animal
tiger live in jungle
tiger be ferocious
accordion produce music
accordion worn on chest
airplane found in airport
airplane involved in crash
tiger used by circus
```

`wheelbarrow be push`
`tiger do roar`

Features need not be true of all instances of the concepts. So `tiger used by circus`, `tiger be ferocious` and `airplane involved in crash` are correct features, even though not all tigers are used by circuses and not all airplanes are involved in crashes.

All noun feature terms have been made singular, so `tiger have tooth` and `accordion have key` would be correct triples, even though tigers have more than one tooth and accordions have more than one key. Similarly, all verb feature terms are in their infinitive form, so `wheelbarrow be push` would be a correct triple, as wheelbarrows can be pushed.

Some features that you will see will be true, and some will be untrue. When judging the correctness of each `<concept> <relation> <feature>` triple, we would like you to select between four possibilities:

c: correct. Triple represents a correct, valid feature (e.g., `tiger be animal`, `airplane found in airport`).

p: plausible. Triple is not correct, but the triple may be plausible in a very specific set of circumstances (e.g., `tiger exhibit dimorphism`, `airplane lands in movie`) and/or the triple may be very general (e.g., `airplane be available`, `airplane be use`), or may be partly correct (e.g., `tiger be black`).

r: wrong, but related. The triple is wrong, but there is some kind of relationship between concept and the relation and/or feature (e.g., `motorcycle be car`, `accordion accompanied by polka`, `accordion plays astronaut`).

w: just completely wrong (e.g., `accordion cremated by Mississippi`, `tiger debunk reputation`, `tiger do squawk`).

Please use your own subjective judgment when making your decisions.