Meronym Extraction from a Biomedical Corpus: A Method for General Corpora Adopted for Biomedical Texts

Bachelor thesis
Niels van Dijk, n.van.dijk@student.rug.nl
Supervised by Jennifer Spenader

Abstract Most research into lexical pattern extraction has been performed on general corpora, but recently research into extracting patterns from biomedical texts has become another topic of interest. I present an algorithm for extracting meronym pairs (part-whole relationship, e.g. finger and hand) from biomedical texts. This algorithm is based upon a method for the extraction of meronym pairs from general corpora. I evaluate the validity of using methods developed for general corpora on more domain-specific texts by comparing the results of the original method with our algorithm. My findings suggest that the type of corpus is not the main factor in performance, but the specific difficulty of using simple methods is.

1 Introduction

Consider the pair of words: (*frontal lobes*, *brain*). In this case, the phrase *frontal lobes* is said to be a meronym of *brain*. A popular way to express this relationship is: for a pair of words (X, Y) X is a part of Y.

Meronymy is one among many possible lexical relationships between words, like synonymy or antonymy. The knowledge of what lexical relationship there is between two words or phrases in a text is useful for many natural language processing tasks and can be valuable in retrieving information from large bodies of text like collections of articles.

In this thesis I present an algorithm that extracts part/whole-pairs from a biomedical corpus. This algorithm is based upon the method used by Berland and Charniak (1999) to extract part/whole-pairs from a corpus consisting of news-wire from several US newspapers. Modifications to their method are made to automate the procedure, remove most supervision, and to accommodate the differences resulting from the different genre of the corpus, e.g. being able to process phrases instead of working only with single words – many entities in biomedical literature consist of more than one word – as Berland and Charniak did.

The core of the method takes pairs of words that are known to have the part/whole relationship and finds sentences in a corpus in which both words of the pair are used. This sentence is then used to extract a pattern from it by treating part of that sentence (in this case the words between the two words of the pair) as a pattern, replacing the words in the pair with place-holders. Patterns found in this way are used to extract new part/whole-pairs by matching them to the sentences in the corpus again. This procedure can then be repeated with the new-found pairs to extract new patterns from the same corpus. Section 2 describes the exact algorithm.

This thesis focuses on the biomedical domain for several reasons. The biomedical domain is one in which many online text resources have recently become available in the form of online lexicons, collections of articles and ontologies for specific topics in the field. To efficiently access these online resources, link these resources and categorize them, natural language processing has become an increasingly important topic for the biomedical domain (Hunter and Cohen, 2006).

This makes the biomedical domain a good place to start in discovering how well lexical information extraction methods translate from the general to a more specific domain. The aim of this theses is to make this comparison for meronyms and identify what influences a more specialized corpus has on the performance a algorithm developed for general corpora.

The remainder of this section contains background information about meronyms, explores the differences between a general corpus and a biomedical one, and gives an overview of lexical pattern extraction in general and more specific meronyms or part/whole-relationships. Previous related work is also discussed.

The next section is dedicated to a description of the algorithm that is developed and explains how the results are obtained. The results are presented in the third section and in the last section I discuss the results and what factors influences them. Finally, I give some suggestions for future research.

1.1 Meronymy and previous work

Meronymy is often denoted as the `part-of` relationship. This relation could be expressed by an object physically being part of another object – (*leg*, *human*) – but more abstract objects are also possible. For example, Winston et al. (1987) defines meronymy as a complex relationship, actually consisting of 6 sub-relationships. Among these the relation between an object and the material it is made of (*plastic*, *bucket*), a portion of a mass (*second*, *hour*), or even a feature of an activity like (*chewing*, *eating*) are all defined. Most research on extracting meronyms from a corpus makes no use of this subdivision in six different relationships.

My algorithm is based on the method of Berland and Charniak (1999), which in turn is adapted from a method introduced by Hearst (1992). Hearst provided the general scheme for the kind of extraction that is used in my algorithm, which is meant as a general method, not specifically for meronyms:

1. Identify a number of 'seeds' (pairs of words that have the relation that is being extracted) and find sentences in a corpus in which both words of a pair are used.

2. Generalize this sentence to a pattern, replacing the words from the seed with place-holders.

3. Match these patterns to sentences in the corpus, returning the words that take the place of the place-holders.

4. Treat these returned words as a new pair, expressing the relation sought after.

The pairs found in step 4 can be used to take the place of the seeds that were manually identified in step 1, finding more patterns and pairs by repeating the procedure. To illustrate this procedure consider the following example:

1. Consider the pair (*tire*,*car*), in which *tire* is a meronym of *car*. We search the corpus for a sentence containing the seed pair, e.g. : *The gun fires three shots at the car's rear tire.*.

2. We generalize this as a pattern to the the found words in the seed pair and the text between them. If we replace the words in our seed with place-holders, this gives: `<WHOLE>`*'s rear* `<PART>`.

3. Part of another sentence in the corpus is: *a car blocking the restaurant's rear door.*, which matches the created pattern. On the position of the place-holders are: *restaurant* for `<WHOLE>` and *door* for `<PART>`.

4. We end up with a new pair expressing the meronymy relation: (*door*, *restaurant*).

Hearst[1] tested her method by applying it to an 8,600,000 word corpus once and matching the results with the lexical database WordNet (Fellbaum, 1998). She reported that in this preliminary experiment good results were achieved for hyponymy (the relation between a member of a class and its class, or the `is a` relationship), but failed to apply the method with success to meronymy, observing that patterns found for meronymy do not tend to uniquely identify it, but can describe other relations as well.

Another observation of Hearst pertaining to this research is the challenge in the generalization of modifiers in patterns in the (bio)medical domain. She states that while in general corpora these modifiers can commonly be generalized or omitted, in the biomedical domain they should be preserved.

Berland and Charniak (1999) decided on a modification of Hearst her approach that presumably[2] performed better on extracting part/whole pairs. Instead of finding pairs that consist of a new part and a new whole, they decided on finding parts for 6 different wholes.

They selected patterns following Hearst by using pairs that express the part/whole relationship and find sentences in which both words of the pair occur. From a list of five patterns constructed by themselves, they chose the two that performed best in a preliminary experiment. Of note is that both patterns used are very generic (high recall, low precision).

They ran these patterns over the LDC North American News Corpus (100,000,000 words) tagged with part of speech (POS) information, but instead of using a place-holder for for the position of the whole, substituted the wholes they were finding parts for. Possible parts were restricted to single nouns, but the patterns made no distinction between singular or plural forms.

The last step of their method consisted of ordering the list of found parts (from high probability that a found part is a real part to a low one). For this ordering they used an advanced statistical method, which they name as a likely reason for their improvement upon Hearst her results. They report 70% accuracy for the top 20 and 55% on the top 50 words. Evaluation of the found pairs was done by majority vote of five informants.

In their error analysis they find no single cause for the errors, observing that POS-tagger mistakes, ambiguous patterns – patterns representing not only meronymy – and sparse data contributed most to the mistakes.

One of the more successful approaches so far in

[1] Note that Hearst did not explicitly used seeds, but gave several options how to "gather a list of terms for which [the lexical relation of interest] is known to hold".

[2] Hearst gives no actual score, so direct comparison is not possible.

the extraction of part/whole pairs (among pairs expressing other relations, like hyponymy and succession) has been described by Girju et al. (2006). They also based their method for identifying patterns on Hearst, but it differs from previous work by taking into account the several different relations described by Winston et al. referred to earlier. They do not create different patterns for the different sub-categories, but place constraints upon the nature of of the words of a pair for patterns, e.g. a pattern might only be applicable to words denote an abstract concept.

Their approach relies heavily on supervision: manual annotation of the training corpora, manual pruning/selection of good patterns. They use WordNet not only to acquire part/whole pairs to use as seeds, but also the information about what the words denote, like the example given before: words could be classified as abstractions, entities, states, psychological features and others.

The extra information used make it possible for Girju et al. to use generic patterns without losing precision: they report good scores (an average precision of 80.95% and recall of 75.91%). Testing was done with two constructed test corpora of each 10,000 words. Successful identification was decided through inter-annotator agreement.

A recurring problem with meronymy is the ambiguity of found patterns: patterns do express the relation of meronymy, but also many other relations, which gives a good recall, but a bad precision. Girju et al. solve the problem of generic patterns by adding more information, but other solutions have been found. Pantel and Pennacchiotti (2006) use their Espresso algorithm with success; their method is developed to harvest semantic patterns in general, but performs well on meronyms as they demonstrate.

The Espresso algorithm uses generic patterns (high recall, low precision) to extract pairs that express the desired relation, but rejects incorrect instances if the found pair is not also instantiated by reliable patterns (high precision). This leads to the second innovation in their work: patterns are assigned not a measure of frequency, but one of reliability. To test if an instance extracted by a generic pattern is a feasible candidate to express the relation they are looking for, they use a large corpus (the world wide web in this case) to see if reliable patterns also occurs with the found pair of words. For meronyms Pantel and Pennacchiotti report 80% precision on a corpus of almost 6,000,000 words.

Pantel and Pennacchiotti's method of deciding the reliability of pattern deserves a special mention, as it has influenced the method for pattern reliability in the algorithm described in this thesis. Like Pantel and Pennacchiotti, I use a measure of well candidate patterns have found the seeds used in the algorithm.

As I have described, the above methods all use pairs to start their method, the so-called seeds. Berland and Charniak use 36 manually picked seeds, 6 for each whole they are extracting parts for; Girju et al. use pairs from words expressing the part-whole relationship in WordNet; and Pantel and Pennacchiotti do not specify how they acquire their seeds.

The question is how much influence these seeds have on the performance of the algorithm, especially considering meronymy might consist of 6 sub-relationships as described by Winston et al.. Ittoo and Bouma (2010) make an interesting observation about these seeds. They test the algorithm of Pantel and Pennacchiotti with a set of heterogeneous seeds (a mix of seeds expressing all 6 sub-relations of meronyms identified by Winston et al.) and a set with all seeds expressing the same sub-relation.

Ittoo and Bouma (2010)'s findings indicate that the homogeneous set does not exclusively let the algorithm find part-whole pairs from the particular relationship expressed and that the heterogeneous set tends to converge to one of the sub-relations of meronymy and confirm that the choice of seeds influences the output much. Their advice is to use seeds from a single category concerning part-whole extraction.

While some research has focused on more specialized corpora instead of general corpora, Roberts (2005) is to our knowledge the only one describing a method for extracting part-whole pairs from biomedical resources. Again his method is based upon Hearst, but fully automated and iterative.

Roberts reports good results: a recall of 73% and a precision of 58%, but uses a corpus that is "highly regularized and, unusually rich in meronyms" (page 54 in (Roberts, 2005)), consisting of biomedical lexicons and ontologies.

Another effort to extract part-whole pairs from a more specialized domain is from Ittoo et al. (2010). In this case the patterns were learned from a general corpus (the English Wikipedia in this case) and then applied to the more specialized corpora, both textual corporate databases. With a precision of 81% the algorithm performance is comparable with the other more recent work described earlier.

The last issue I want to mention is that pertaining to the differences between general corpora and more specialized ones, especially biomedical corpora. Not much work is published indicating explicit differences, but Ittoo et al. (2010) have tried a method based on Pantel and Pennacchiotti their Espresso algorithm to measure performance on more specialized corpora.

Their results indicate that this general method works well on more specialized corpora (in this case 143,255 textual narratives of customer complaints

and repair actions): an overall precision of 81%. Of note is that the extraction of the patterns was done on another, general corpus, namely the English Wikipedia.

An observation more relevant to specifically the biomedical domain is that the many abbreviations and synonyms often make natural language processing tasks more complicated (Aronson, 2001). Another attribute of biomedical texts is that the phrases of interest (in this case possible meronyms) often have modifiers that cannot be dismissed as in other corpora (Hearst, 1992).

The goal in this thesis is to adapt the methods generally used on general corpora to work on a biomedical corpus. The identification of our patterns is based on the general method Hearst introduced for and incorporate it into an algorithm that is iterative and unsupervised, much like the method Pantel and Pennacchiotti present. The algorithm will output an ordered list of possible part/whole pairs given a corpus and a number of examples of meronymy (the seeds).

The method differs from an automated version of Berland and Charniak in that this method is not limited to certain wholes and that it allows for noun phrases, which given the nature the used corpus is a requirement. The creation and selecting of the patterns to use is fully automated, like with Roberts, but the corpus used consists of articles rather than lexicons and ontologies. Even though Girju et al. report a good performance, the goal is an automatized and low-cost algorithm, not an heavily supervised one.

Next to performance the interest is mainly in the implications of using a method used on general corpora on a specialized corpus, how do the differences influence the performance. We test different ways of ordering the pairs.

2 Methods

2.1 Corpus

The corpus used to extract part/whole pairs from is the BioMed Central's open access full-text corpus[3], consisting of 99878 articles of biomedical research. Some preprocessing is necessary, mainly adding part of speech information, for which MedPost[4] was used (Smith et al., 2004), which is specialized in POS-tagging of biomedical text. But also the deletion of functional code that links to figures, tables, etcetera.

The choice to only add part of speech information is motivated by the desire to keep the algorithm as broadly applicable as possible. POS-tagging is cheap computationally speaking and adds useful information to generalize the patterns and identify the pair- and whole noun phrases the algorithm extracts. The availability of high quality POS-taggers that work on natural language texts facilitates this matter.

After preprocessing the corpus consisted of $92,763,401$ words (including interpunction and numbers) spread over $4,525,911$ lines, with on each line one sentence. Some anomalies surfaced during execution of the algorithm, like long uninterrupted sequences of numbers, which the algorithm disregarded.

2.2 Seeds

The algorithm is dependant on seeds – pairs of words expressing the part-whole relationship – to find the right patterns. The choice of the seeds is the only part of the algorithm that is done manually. The literature did not suggest what pairs would yield good results, as the only research that also extracted meronyms used an alternative way to generate good patterns or did not mention the seeds used.

From the research of Pantel and Pennacchiotti and Ittoo et al. it becomes clear that the choice of seeds play an important role; it influences what kind of relationships the algorithm will find (Ittoo et al., 2010) and the seeds will play a crucial role in the quality of the patterns extracted and in turn the pairs extracted by those patterns.

I have chosen to pick the seeds manually; the lack of research on this specific topic made it difficult to pick the seeds in a more principled way. Preliminary experiments with the algorithm were used to identify seeds that were sufficiently represented in the corpus and expressed the meronymy relationship well.

The research of Ittoo et al. recommends using a set of seeds representing the same form of meronymy over a mixed set where different subrelations of meronymy are represented. I chose to this recommendation and used seeds which are from the – probably most straightforward – relationship that represents a component and the object it belongs to (Winston et al. (1987) calls this the `COMPONENT-INTEGRAL OBJECT` relationship). The seeds I use are listed in table 1.

2.3 Implementation

Due to the size of the corpus and data generated the algorithm is implemented in `C++` to keep the time needed to run the algorithm needs low. To further decrease running time I used string matching instead of regular expressions, which gives a very significant performance increase.

[3] `http://www.biomedcentral.com/info/about/datamining/`

[4] `http://www.ncbi.nlm.nih.gov/staff/lsmith/MedPost.html`

Part	Whole
C-terminal ends	proteins
3'UTR	mRNA
ribosomes	cell
chromosome	genome
nucleus	cell
cytoplasm	cell
race	species
node	network
node	tree

Table 1: This table lists the seeds I have used for the algorithm. The algorithm automatically generates a singular and plural version of all pairs found and the seeds.

2.4 Algorithm outline

The general outline of the algorithm consists of the following steps:

1. Take seed pairs and find sentences in which both words of the pair occur. Store the part of the sentence between the two found words as a pattern with place-holders `<PART>` and `<WHOLE>` in the appropriate places. Store the part-whole pair with the pattern.

2. Evaluate the list with found patterns. Generalize each pattern by removing non-relevant information and then remove all duplicate patterns, retaining what pairs were used in finding the pattern. Sort the list of patterns by a metric describing the quality of the pattern (fitness).

3. Take the X best patterns and match them with sentences in the corpus. If the pattern applies to a pattern, add the found pair to the list of pairs associated with the pattern.

4. Calculate the fitness of the patterns again now that new information is added by finding pairs. Prune patterns that are extremely productive or extremely unproductive.

5. Combine all the pairs associated with the found patterns. Remove duplicates (retaining information about how many times the pairs occurred). Prune all pairs that occur only once in the whole corpus and assign a metric to each that represents its quality. Take the best X pairs.

6. Repeat step 1 through 4 with the pairs found in step 4 until no new pairs are found.

7. Return a list of pairs that are sorted by the probability the pairs have the part-whole relation.

In the first step of the first iteration, the pairs normally generated in step 4 are replaced by the seeds provided to the algorithm. In selecting the patterns to find pairs with and pairs to find patterns with, the algorithm chooses only patterns and pairs that were not used before; already used patterns would not yield new pairs or patterns and would distort the measuring of their quality.

As the algorithm needs to make a comparison between lists of found pairs, the algorithm will always go through more than one iteration. The second iteration will provide a list of pairs that can be compared to the results from the first iteration.

To decide if a new iteration yields new pairs, I choose not to compare the whole list with pairs, but rather to compare the top 50 of the found pairs and check if this list has changed compared to the previous iteration. The reason for this is that the lists are sorted and if the whole list would be compared, the decision to run another iteration would be made on the grounds of presumably bad pairs, while the interest lies with the good pairs.

It might seem odd to calculate the fitness of the patterns twice. This is done because after finding the pairs we have new information about the behaviour of the patterns in terms of their productivity. As I explain later in this section the fitness of the patterns is also used in assigning the pairs a fitness. The quality of the pattern ranking will influence the quality of the ranking of the pairs.

2.5 Patterns

In the previous subsection I describe how the algorithm takes the text between a pair and a whole and used it as a pattern. This is not without constraints. The length of the sentences might make very specific and non-productive patterns and the assumption of proximity is a reasonable one.

A possible pattern is defined as a part and a whole (not necessarily in that order) with up to three elements between them, but at least two:

```
<PART> Element Element (Element) <WHOLE>
```

An element, pair, or whole can be a word, a noun phrase (defined as any number of nouns[5] possibly preceded by an adjective). Elements have the added property that they can also be interpunction or the description of a category. This last provides a generalization of the pattern by excluding information that is probably irrelevant, like an exact number instead of that there can be any number. Numbers are always generalized, but I chose not to generalize determiners, observing that patterns to extract part-whole pairs are often quite generic already and productivity of patterns is not a goal.

[5]We follow Berland and Charniak (1999) in the use of nouns/noun phrases, though their algorithm only worked with single nouns and not multi-word terms.

About the length of the pattern: some prelimenary testing was done with shorter and longer allowed patterns. Longer patterns never showed up in our testing runs, being dismissed by the algorithm for occuring not often enough. In some tests, shorter patterns were extracted, but influenced the results in a bad way due to being to generic.

No more generalization then this is applied in the patterns. As I described in the introduction, previous research observed that patterns used to identify meronymy are ambiguous and making them less specific might increase this property. Recall that Pantel and Pennacchiotti used reliable patterns (high precision, low recall) to validate instances found by generic ones (low precision, high recall).

Even after removing duplicates the algorithm finds too many patterns to use them all to find new part-whole pairs, and a measure of how well the pattern performs is necessary. Not only to keep the time to execute the algorithm within bounds (by using only the best patterns), but low quality patterns will degrade the performance.

The only information available at the time of the first ranking pertaining to how well the patterns find meronyms, is how well they find the different seeds/found pairs. While each pattern is the direct result of at least one of the presumed part/whole pairs, the measure of how well it finds all the different seeds would indicate it predicts a part-whole relationship:

$$\text{seed fitness} = \left(1 - \frac{\text{seeds}_{\text{dif}}}{\text{seeds}_{\text{tot}}}\right) * \frac{\text{seeds}_{\text{tot}}}{\text{seeds}_{\text{pos}}}$$

Here, $\text{seeds}_{\text{tot}}$ is the total number of seeds found in the pattern; $\text{seeds}_{\text{dif}}$ the number of different seeds; and $\text{seeds}_{\text{pos}}$ the highest possible number of seeds that a pattern has found.

Another property that is desirable for a pattern is how specific it is. A pattern that finds many different pairs (but not necessarily part-whole pairs) but only a few of all of those pairs is probably less useful for the purpose of finding similar words as the pair(s) that created it than a pattern that finds fewer different pairs but a lot of each. The value for this fitness is calculated in an analogous way to the seed fitness.

The measures described above are combined to describe the quality of the pattern and patterns are ranked accordingly. In this thesis I chose to treat meronymy as a simple relation and not the complex relation described in section 1. This assumption underlies both factors that decide the quality of a pattern, but in lieu of more information serves better than no ranking.

The extremities of over- or under-production are weeded out from the list of found patterns instead of relying only on the fitness measure. Patterns that produce more than 100,000 pairs if searched for in the corpus, or patterns that found only one different pair are removed from the list with found patterns. A last allowance I made is to exclude certain words that are in general very frequent in the corpus, like *table*, *figure*, *method* and others. These words occured so frequently and matched the patterns for meronymy so often, that retaining them made extraction of part/whole pairs impossible.

2.6 Part-whole pairs

Part-whole pairs consist of two noun phrases (again defined as any number of nouns, preceded by an adjective). When searching the corpus for pairs or comparing pairs with each other the grammatical number of the nouns is never taken into account.

I placed some constraints on which nouns or noun phrases are accepted as member of a part/whole pair. Pairs cannot contain special signs or interpunction ('#', '$', '%', '(' to give some examples) and have to be longer than 1 letter[6]. I follow Berland and Charniak by omitting words that with suffixes like in '-ity', '-ness' and 'ing', as these often indicate qualities rather than entities.

When searching the corpus for new part-whole pairs using a pattern the filling of the place-holders will be greedy: the longest possible noun-phrase will be assigned to the place-holders in the patterns. This brings the risk of unwanted modifiers, but the presence of many modifiers that are part of the presumed parts and wholes seem to justify this in contrast with omitting them.

As described in algorithm outline, the final step of each iteration aims to keep only the possible part-whole pairs that we deem fit. This step is necessary to keep producing relevant patterns and because it is not feasible to use all pairs in the next iteration.

To create suitable pairs for the next iteration or possibly for output, a list is composed of the pairs each pattern has found. This list is shortened by removing all duplicate pairs, as a pair might of course occur in more than one pattern. The information about how many times the pair was found in a pattern in the corpus is of course retained.

The second step is to assign a measure to each pair representing the probability it expresses the part-whole relation (fitness). This is again a combination of several factors. The first is how many times it is found in the corpus by the patterns. This measure might disregard pairs that are actually part-wholes, but the algorithm needs a fair number of instances of each pair to perform well.

The second measure comes from how many different patterns it occurs in and how well spread it is between the different patterns. For example, a pair

[6] These choices were mainly made to exclude anomalies in the corpus.

gets a better score if it appeared 8 times in each different pattern, than 2 times in one many different patterns and 60 times in one. The idea behind this is that with the ambiguous nature of the patterns the appearance of a pair in several different patterns is a reliable indicator of meronymy. This measure is weighted by the fitness the patterns are assigned. This idea is originally proposed by Jones (2002).

2.7 Evaluation

An automated way of testing each of the pairs in our output is not feasible, as systems like WordNet (for general corpora) are not yet easily available for biomedical texts and the smaller efforts that have been done for such systems in the domain are too specialized; one of the goals of this research is to actually to help to improve such systems.

Measuring the performance will be done by manual inspection of the output list by one judge (the author). While the pattern ranking has been decided by preliminary experiments, we will present results of runs with different ways of ranking the pairs (which influences the seeds created for next iterations as well as the output of the algorithm). We run the algorithm 5 times. Recall that the fitness of seeds is a combination of:

- How many different patterns it occurs in and how often.
- Frequency of the pair in the the patterns used.

Each run we assign a weight A and B to these 2 measures, with $A+B=1$. We variate these weights to go from a situation where only the first measure decides the fitness to a situation where only the second measure decides the fitness in 5 steps.

While not part of the official results, the patterns extracted and used to find new pairs offer insight in the workings of the algorithm and they will be presented in the next section along with the lists of pairs.

3 Results

The algorithm was run several times to decide on the best way of ranking the pairs. In the previous section the stopping condition of the algorithm is described as: stop running if the list of top 50 pairs (the pairs that were were most likely to express the meronymy relation) does not change compared with the previous iteration. In every run of the algorithm this resulted in 2 iterations before the algorithm stopped.

The first table with results, table 2 on page 7, gives an overview of the results with different ways of ranking of the pairs: only how many times a pair occurs, only how well the pair is represented among the different patterns, and a combination of these two measures in several gradations. The table shows that the best results are acquired when the measure of how well the pairs are acquired has the most weight. The run which I use as the definitive run is the second one, with $A = 0.75$ and $B = 0.25$. All following tables give more specific results from that run. A and B stand for the the weighting factor: the measure for pair spread and the measure for pair occurrence (remember, both are in $\langle 0, 1\rangle$) are multiplied by A and B and then added together. To keep the end result of this also in $\langle 0, 1\rangle$, A and B always add up to 1.

Weights$\langle 0,1\rangle$		Accuracy (%)		
A	B	10	20	50
0.00	**1.00**	**60**	**35**	**28**
0.25	**0.75**	**60**	**35**	**28**
0.50	0.50	60	35	24
0.75	0.25	60	35	24
1.00	0.00	30	30	26

Table 2: This table gives an overview of 4 runs of the algorithm, each with different weights for the measures that that make up the value describing the 'fitness' of a pair: which should indicate how likely it is a pair is a part-whole pair. A is the measure of how well it is represented among the different patterns; B indicates frequency in the corpus. The numbers under A and B are the weights that are giving to these values. All values used are normalised to fall in$< 0..1 >$.

Table 3 on page 8 gives an overview of the top 50 pairs resulting from the run with the best way of pair ranking (according to the results found in table 2). Pairs that are judged to qualify as part-whole pairs are printed in bold, the ranking is the same as the actual result, as can be observed from the Q score. All patterns used in this run can be found in table 4 on page 9. The last table I present here gives all the seeds used. The seeds listed in table 5 contains again the seeds presented in the previous section and the newly generated ones.

4 Discussion

The list with found pairs makes it clear that the algorithm performs not very well; it reaches an accuracy of 35% for the first 50 pairs, though the score of 60% for the first 10 pairs is acceptable. The scores from table 2 indicate that the ranking of pairs has some effect by concentrating most of the positively identified part-whole pairs in the top 10, but the total performance is not enough to contribute to the furthering of online lexical resources in the biomed-

Part	Whole	Q	#
gene	**array**	0.882	153
probe	**array**	0.482	73
spot	**array**	0.474	70
gene	**microarray**	0.467	70
type diabetes	patient	0.444	66
gene	**chromosome**	0.437	64
previous studie	agreement	0.395	56
probe	**microarray**	0.393	55
ml	time	0.39	55
mean	normal distribution	0.39	55
COPD	patient	0.365	50
effect	expression	0.356	46
gene	basis	0.336	42
previous report	agreement	0.32	41
light	mechanism	0.306	36
studie	effect	0.286	32
light	evolution	0.281	31
group	basis	0.271	29
gene	**chip**	0.271	29
information	number	0.271	29
comment	manuscript	0.265	28
marker	chromosome	0.256	28
type diabetes	people	0.255	28
limit	number	0.255	26
gene	**X chromosome**	0.255	26
gene	list	0.255	26
effect	number	0.255	26
feature	**array**	0.255	26
spot	**microarray**	0.25	25
effect	level	0.24	23
protein	basis	0.24	23
position	chromosome	0.24	23
information	gene	0.24	23
effect	performance	0.235	22
asthma	patient	0.23	23
probe	**chip**	0.219	19
location	genome	0.219	19
ng	treatment	0.214	20
probeset	**array**	0.214	18
light	question	0.209	17
pressure	tracheal wall	0.209	17
light	role	0.209	17
present	array	0.209	17
nodes	**graph**	0.209	17
studies	role	0.203	16
information	nature	0.203	16
information	patient	0.203	16
information	distribution	0.203	16
gene	**genome**	0.202	18
SNP	chromosome	0.202	18

Table 3: In this table I list the 50 pairs the algorithm deemed most likely to have the part-whole relationship. Q denotes the probability of the pair being a part-whole pair and # denoting the number of times the pair was found in the corpus by the used patterns. Bold pairs are judged to be part-whole pairs by manual inspection.

Pattern		Q	#
	First iteration		
P1	<PART> of the <WHOLE>	0.99	459
P2	<PART> in the <WHOLE>	0.99	449
P3	<WHOLE> with (number) <PART>	0.84	50
P4	<WHOLE> , the <PART>	0.81	52
P5	<PART> on a <WHOLE>	0.80	26
P6	<PART> within the <WHOLE>	0.77	27
	Second iteration		
P7	<PART> on the <WHOLE>	0.99	510
P8	<WHOLE> with (number) <PART>	0.92	320
P9	<PART> of (number) <WHOLE>	0.89	231
P10	<PART> in a (number) <WHOLE>	0.89	213
P11	<WHOLE> for a <PART>	0.89	202
P12	<PART> included in the <WHOLE>	0.89	203

Table 4: In this table I present the patterns the algorithm has extracted from the corpus and used to discover new part-whole pairs. The pattern is listed first, Q gives the quality of the pattern and # how many times the pattern occurred for the relevant pairs. The format of the patterns is copied directly from the algorithm. The codes connected with an underscore are added by the Part-of-speech-tagger and denote the the category of the words. Those words tagged with 'MC' (the code for a number) are always generalized.

Part	Whole	Q	#
	First iteration		
C-terminal ends	proteins	NA	46
3'UTR	mRNA	NA	57
ribosomes	cell	NA	28
chromosome	genome	NA	491
nucleus	cell	NA	279
cytoplasm	cell	NA	234
race	species	NA	23
node	network	NA	645
node	tree	NA	697
	Second iteration		
genes	array	0.88	3703
COPD	patients	0.37	232
CF mice	B6 background	0.15	9
AD	patients	0.13	113
13-kb gene cluster	chromosome	0.08	2
2′-methyl-MPTP injections	same day	0.08	2

Table 5: In this table all the seeds used in the algorithm run are listed. The seeds listed under the first iteration are of course identical with those mention in section 2, table 1. The seeds from the second iteration are generated by the algorithm: the 6 top ranked pairs after one iteration. Like patterns, each pair has been given a measure of its quality, again denoted with Q. This value lies $\langle 0, 1 \rangle$. For the first run the seeds could not be given such a value, hence the notation 'NA'.

ical domain.

The idea behind the algorithm is to find patterns that express the relation the seeds have, find new pairs with these patterns that should express the relation the pattern have and repeat the procedure. This observation makes clear that there are several points where the algorithm might function less than optimal.

Every step in the algorithm its execution will influence all steps that come after it. If one step fails – for example if an iteration returns a list of pairs that do not express the part-whole relationship or not exclusively enough – the search for new patterns will be influenced, returning patterns that do not express the part-whole relationship.

I will start this discussion by going through the results with the changing pair ranking and combine that with going through the best of the runs and analysing its workings with help from the used patterns and pairs. Our original goal involved also an analysis of the difference a specific domain corpus makes as opposed to a general one, but as the algorithm is not on par with comparable algorithms in performance, I will have difficulty discerning failure in the algorithm from peculiarities in the corpus. Still, there is some opportunity for observations about the specific corpus used, which I will make later this section.

4.1 Workings of the algorithm

As described in section 2 I chose all seeds from the `COMPONENT-INTEGRAL OBJECT` relation, rather than the other sub-relationships, which are more abstract. Ittoo and Bouma (2010) recommended using one type of sub-relation and their observation for their algorithm was that other sub-relations would also manifest in the end results. This seems not to be the case with our algorithm. But before we get to the final results, first the patterns generated in the first iteration.

Looking in table 4 at the patterns generated in the first iteration, I suspect the patterns generated are generic, but good at expressing the part-whole relationship. Recall that these patterns are not only ranked by their frequency, but more importantly by how many different seeds each pattern has found. Several of these patterns are found in other literature on meronym extraction, for example, Berland and Charniak (1999) mention $P1$ and Girju et al. (2006) $P2$ (also noticing it for its genericness).

If we look at the pairs that are found and ranked highest in the first iteration after extracting them using the patterns, it becomes clear the patterns are too generic or the ranking of the pairs needs to be improved: most of the found pairs do not express meronymy. As these pairs are used to extract new patterns in iteration 2 and the quality of the seeds influences the quality of the patterns, we expect a decrease in the quality of the patterns. This is exactly what happens (as table 4 shows). While pattern $P7$ through $P10$ seem generic but still might express meronymy, pattern $P11$ and $P12$ I would dismiss.

The results with the different ways of ranking the pairs (table 2) are not much different, and while I did not print all the results from the different runs, I have inspected them and noticed little difference in what patterns were found, which pairs were used as seeds and which pairs were on the output lists. Generally they only differed slightly in order, patterns as well as pairs. This could indicate that both measures work about equally or that another factor is more powerful than the pair ranking, that factor being the strength of the patterns.

Concerning the corpus and more specifically that it consists of biomedical texts we can make a few observations about the patterns and pairs used during the execution of the algorithm. The patterns extracted from the corpus contain no indication that they are extracted from biomedical texts, insofar I can discern; no specific words or phrases are used. What does stand out is that 4 out of 12 patterns contain a number, which is not representative of what examples of found patterns other research mentions. This may be due to the biomedical nature of the text, but this might also be a result of the texts consisting of academic writing.

The found pairs are another matter. The pairs used as seeds (not considering the manually picked initial seeds) are exclusively made up of biomedical terms. The final list of the top 50 ranked pairs follows this trend, but contains some other pairs. These other pairs are recognizable as terms you would expect in academic writing. Pairs from categories other than biomedical or academic writing miss from the list. I see two possibilities:

1. The initial seeds were mostly from the biomedical domain, with a few that one would expect in academic writing, this might have influenced the kind of words the algorithm finds.

2. Biomedical/academical terms that the patterns extract outnumber general terms by far. A full inspection of all pairs picked could be done to check this. As the algorithm extract millions of terms this is not within the scope of this thesis.

The first option seems implausible, as the patterns extracted are comparable to patterns used by Berland and Charniak and Girju et al. on general corpora. This leaves the second option as the likely reason. I proceed now with a closer look at the end results.

4.2 Found pairs

The output of the algorithm consists of an ordered list of pairs (see section 3 or table 3 on page 8). Out of all of the 50 pairs I identify 13 pairs that express the part-whole relation. Many of the pairs that do not express meronymy contain more abstract terms taken from the academic domain. Examples are: *previous study*, *previous report*, *effect*, *performance*. The presence of these terms is probably brought about by the frequency of the words in the text in combination with fitting several of the patterns also expressing meronymy.

I also notice that the results contain very few noun phrases, nearly all words are single word terms. As the algorithm allows for noun phrases and does nothing to promote or restrain them, this result seems to oppose Hearst notion that modifiers are very important in medical texts. Perhaps her observation will be more true for even more specialized texts from the anatomical domain. I have done some preliminary experiments using solely seeds consisting of phrases rather than words, without any noticeable difference in the phrase/single word ratio of the output.

A possible cause of this is the greedy noun phrase recognition the algorithm uses. The algorithm will always take the longest noun-phrase possible to fill place-holders for part and whole in a pattern. The fact that I make no difference in meaningful modifiers in phrases (for example *superior colliculus*) versus modifiers that are not part of an entity (for example *mature superior colliculus*) can mean that phrases that should be considered the same as in the examples given, are identified as two different words. This will influence decrease their fitness.

4.3 Conclusion

In section 1 we introduced several more or less successful methods for extracting part-whole pairs by extracting patterns. All research described there used a method to circumvent the problem of these patterns being generic or ambiguous, except for Hearst (1992), who reported no success in extracting part-whole pairs.

Berland and Charniak (1999) only find pairs for given wholes and their method is not iterative; Girju et al. (2006) use manual selection and annotation; Pantel and Pennacchiotti (2006) introduced the Espresso algorithm and control generic patterns by using reliable patterns to check the validity of an instance found by a generic pattern; and Roberts (2005) confines himself to a very specific corpus.

The algorithm presented in this thesis lacks a reliable option to keep the generic patterns in check. The patterns found in the first iteration are indeed expressing meronymy, but mostly as one relation among others. This in turn influences what pairs are found and used as seeds for the next iteration, making the algorithm 'spin out of control'. Indeed, compared to other research the algorithm is mostly like an automated version of Hearst, which reported no great success.

Several preliminary experiments were done to see if this phenomenon could be countered, such as in increase in seeds (which yielded the same patterns, as these patterns were generic, but represented meronymy well), different ways of deciding what the best patterns are, and pruning overproductive patterns. All measures did not improve the results of the algorithm.

I suspect the problem lies not with the biomedical corpus we used, but with the lack of a method to keep the generic patterns in check. Looking at the pairs the algorithm found, distortion seems to come more from academic writing, than biomedical terms. While using this information to for example disregard typical phrases found in academic texts, the genericness of the patterns would probably still prevent good results.

With a newly created algorithm, untested on a general corpus, it is not easy to make a good comparison between general and more specialized corpora:

- Found words were solely from the biomedical or academical domain. More general phrases are either missing or not well represented.
- We found patterns comparable to patterns found in methods for general corpora despite finding specialized terms. One might suspect the general patterns carry over to more specialized domains. This is consistent with the findings of Ittoo et al. (2010), who extracted patterns from a general corpora and used those on a specialized corpus with success.
- Phrases as opposed to single words are not especially common in this biomedical corpus. Given the scope of the corpus used this might be true for non-further specialized biomedical corpora in general. As most recent research allows for phrases this is not a finding of great impact.
- The problem of generic patterns will carry over from general to biomedical (natural language) corpora.

4.4 Recommendations for future research

This thesis has not answered all questions about how well methods made for general corpora will perform on general corpora. For future research a method that has proven itself on general corpora

would be a better candidate to test how well it performs on more specialized corpora.

The algorithm itself has some scope for improvement. Patterns could be extended to make use of more than only the words between the part and the whole to create more specific patterns that also used words to the sides. This brings other difficulties though and would require more elaborate generalization. To better identify the terms of interest named entity recognition could be applied, which eliminates the need of 'guessing' the relevant phrase boundaries and could improve the extracting of multi-word phrases.

The method could also be adapted to use the Espresso method from Pantel and Pennacchiotti (2006), the framework is already there. It would be interesting to see if using Google, as they did, could also serve for a biomedical corpus. Another way of battling the genericness could be to follow Berland and Charniak and look for parts of given wholes only. This might be less insightful to learn about the differences between general and biomedical corpora, but seems a good way of extending lexical knowledge resources in the biomedical world.

References

Aronson, A. (2001). Effective mapping of biomedical text to the umls metathesaurus: the metamap program. In *Proceedings of the AMIA Symposium*, page 17. American Medical Informatics Association.

Berland, M. and Charniak, E. (1999). Finding parts in very large corpora. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 57–64. Association for Computational Linguistics.

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

Girju, R., Badulescu, A., and Moldovan, D. (2006). Automatic discovery of part-whole relations. *Computational Linguistics*, 32(1):83–135.

Hearst, M. (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.

Hunter, L. and Cohen, K. (2006). Biomedical language processing: Perspective whatŠs beyond pubmed? *Molecular cell*, 21(5):589.

Ittoo, A. and Bouma, G. (2010). On learning subtypes of the part-whole relation: do not mix your seeds. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1328–1336. Association for Computational Linguistics.

Ittoo, A., Bouma, G., Maruster, L., and Wortmann, H. (2010). Extracting meronymy relationships from domain-specific, textual corporate databases. In *Proceedings of the Natural language processing and information systems, and 15th international conference on Applications of natural language to information systems*, pages 48–59. Springer-Verlag.

Pantel, P. and Pennacchiotti, M. (2006). Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 113–120. Association for Computational Linguistics.

Roberts, A. (2005). Learning meronyms from biomedical text. In *Proceedings of the ACL Student Research Workshop*, pages 49–54. Association for Computational Linguistics.

Smith, L., Rindflesch, T., and Wilbur, W. (2004). MedPost: a part-of-speech tagger for bioMedical text. *Bioinformatics*, 20(14):2320–2321.

Winston, M., Chaffin, R., and Herrmann, D. (1987). A taxonomy of part-whole relations**. *Cognitive science*, 11(4):417–444.