Exploiting Multiply Annotated Corpora in Biomedical Information Extraction Tasks

Barry Haddow and Beatrice Alex

School of Informatics
University of Edinburgh
2 Buccleuch Place
Edinburgh, EH8 9LW
Scotland, UK
{bhaddow,balex}@inf.ed.ac.uk

Abstract

This paper discusses the problem of utilising multiply annotated data in training biomedical information extraction systems. Two corpora, annotated with entities and relations, and containing a number of multiply annotated documents, are used to train named entity recognition and relation extraction systems. Several methods of automatically combining the multiple annotations to produce a single annotation are compared, but none produces better results than simply picking one of the annotated versions at random. It is also shown that adding extra singly annotated documents produces faster performance gains than adding extra multiply annotated documents.

1 Introduction

When annotating a corpus for an information extraction (IE) task, it is normal to annotate at least part of it multiple times in order to measure inter-annotator agreement (IAA) and thereby monitor the quality of the annotation and quantify the difficulty of the task. These multiply annotated documents can then be reconciled to produce a gold standard which is used to train and test a machine-learning based IE system. In our experience in the biomedical domain, however, this reconciliation process is very time-consuming, and therefore costly, with the reconciliation of a pair of annotated documents taking much longer than annotating the document for a third time. But if reconciliation is too expensive, and the corpus is to be used as training data, can one make better use of the multiply annotated data?

This paper will set out to answer two questions with empirical evidence presented from biomedical IE tasks. The first question is how best to use the multiply annotated data in training machine learning systems; whether to try to automatically reconcile the documents using some type of combination algorithm, or whether to provide all copies to the machine learning algorithm and leave it to "reconcile" the documents. The second question concerns how best to spend annotation budget, on multiple annotation of documents already in the corpus, or on annotating new documents. To the best of our knowledge, these questions have not previously been addressed in the literature.

This paper is organised as follows: after reviewing related work in the following section, the corpora and experimental system are described in Section 3. An account of the experiments performed is provided in Section 4, with the results presented in Section 5 and discussed in Section 6.

2 Background

The experiments described in this papers make use of a text mining pipeline which was developed as part of the TXM project and integrated into a biomedical curation tool to enable assisted curation for biomedical literature (Alex et al., 2008). In order to train the machine learning components of the system, and to test the system, a collection of full-text biomedical articles were annotated with items of biomedical interest. The decision to annotate full-text articles instead of abstracts as training data for a biomedical text mining system is supported by previous work of Shah et al. (2003) and McIntosh and Curran (2007) who argue that paper abstracts do not always contain sufficient information.

In the pilot data annotation phase of TXM, multiply annotated papers were reconciled to create one gold standard annotation for each paper. However, this process was found to be very time-consuming, particularly as dealing with full-text articles. In order to obtain good machine learning results, a further decision was made to invest valuable annotation time to annotate more papers rather than reconcile multiple annotations. Consequently, the two corpora that were finally annotated as training material for the TXM pipeline were not reconciled and therefore contain multiple annotations for a proportion of papers (see Section 3). These multiple annotations were used to calculate IAA in order to monitor annotation consistency over time.

The collection of IAA during annotation has also been advocated by Lu et al. (2006) and Inderjeet et al. (2005). Furthermore, annotation consistency is an aspect that was highlighted as important in experiments carried out by Alex et al. (2006). Measures of IAA are also used to get a better understanding of how difficult it is to extract such information automatically.

3 Corpora and System

The two corpora used in the experiments in this paper address protein-protein interactions (the PPI corpus) and tissue expression (the TE corpus), and consist of full-text biomedical papers from PubMed and PubMedCentral. The PPI corpus contains approximately 75,000 sentences and the TE corpus around 63,000.

A total of nine qualified biologists were employed to annotate each corpus with named entities and relations (as shown in Tables 1 and 2), to map selected entity types to

identifiers in appropriate standard databases, and to enrich the relations with certain properties and attributes (Haddow and Matthews, 2007). Only the entities and relations will be considered in this paper. Each corpus was split into three sections (TRAIN, DEVTEST and TEST) with the intention that TRAIN and DEVTEST would be used to develop the NLP components, and TEST would only be used for testing the final system.

Type	Count
CellLine	7,676
Complex	7,668
DrugCompound	11,886
ExperimentalMethod	15,311
Fragment	13,412
Fusion	4,344
Modification	6,706
Mutant	4,829
Protein	88,607

(a)

Type	Count
Complex	4,033
DevelopmentalStage	1,754
Disease	2,432
DrugCompound	16,131
ExperimentalMethod	9,803
Fragment	4,466
Fusion	1,459
GOMOP	4,647
Gene	12,059
mRNAcDNA	8,446
Mutant	1,607
Protein	60,782
Tissue	36,029

(b)

Table 1: The entity types and occurrence counts for (a) PPI and (b) TE. Note that *GOMOP* stands for *"Gene or mRNAcDNA or Protein"* and was used when the annotators felt the author employed the term in an ambiguous way.

During the annotation of each corpus, a selection of papers were doubly or triply annotated, as shown in Table 3, and these multiply annotated papers were used to compute IAA. For each pair of corresponding annotations, the IAA is calculated by taking one of the annotators as the gold standard and scoring the other annotator against them by calculating precision, recall and F_1 in the usual way. The F_1 is not affected by the ordering of the annotators in this calculation (Brants, 2000). Entities are compared using exact match as in Tjong Kim Sang and De Meulder (2003), and relations are considered equal if they have the same type and arguments. When scoring relations, only those where both annotators agree on the entities were considered, in order for relation IAA not to be affected by entity IAA. A *combined* IAA is also included, where pairs of annotated documents were scored on all relations, not just those where the annotators agree on the entities. This combined score indicates the level of agreement on the whole task of relation markup. To produce an average IAA for a particular relation or entity type, the F_1 scores on each pair of corresponding annotated papers were micro-averaged, i.e. each relation or entity instance was given equal weight in computing the overall scores shown in Table 4.

Corpus	Annotated item	IAA
PPI	entities	84.9
PPI	relations	76.1
PPI	combined	59.7
TE	entities	83.8
TE	relations	74.1
PPI	combined	55.7

Table 4: Inter-annotator agreement for entities, relations, and the combined entity/relation annotation task.

Both named entity recognition (NER) and relation extraction (RE) were implemented using maximum entropy based machine learning approaches as part of an information extraction pipeline (Alex et al., 2008). The pipeline contains a pre-processing component which performs tokenisation and sentence splitting, linguistic analysis (such as part-of-speech tagging and chunking) and adds some biological markup (indicating species words and abbreviation definitions). The NER component uses the Curran and Clark (2003) named entity tagger augmented with extra orthographic and gazetteer features tailored to the domain, and more fully described in Alex et al. (2007). The RE component uses maximum entropy to classify candidate relations, based on features derived from the context of the entities in the candidate, the text in between, the part-of-speech and chunk tags, the entities themselves, and features derived from indicator words. The latter are either interaction words or expression level words collected from the training data which were marked up by the annotators in addition to the entity and relation annotations discussed earlier. An earlier version of the RE component was described by Nielsen (2006), but has since been extended to address further relation types, and some inter-sentential relations.

4 Experiments

To assess how different methods of combining multiply annotated documents affect performance, models were trained and tested using each of the combination methods listed below. Tests were conducted for both NER and RE, on the two corpora, giving four different testing configurations. Performance measurements are for the DEVTEST and TEST sections of each corpus, which include multiply annotated documents. It was thought fairest to leave the multiply annotated documents in the evaluation set, in order to ensure that it was as large as possible, and so that the system would obtain partial credit for predicting an entity or relation chosen by at least one annotator, even though this means that the system cannot obtain 100% accuracy. The following methods of combining multiply annotated training data were used:

Corpus	Type	Count	Description
PPI	PPI	11,523	Indicates that the text is referring to an interaction between Proteins, Fragments, Mutants, Complexes or Fusions.
PPI	FRAG	16,002	Connects Fragment or Mutant to its parent Protein.
TE	TE	12,426	Links a gene or gene product to a Tissue, indicating that the text is stating that the gene or gene product is expressed in that Tissue.
TE	FRAG	4,735	Plays the same role as in PPI.

Table 2: The relation types in the corpora

Annotations	PPI			TE		
	TRAIN	DEVTEST	TEST	TRAIN	DEVTEST	TEST
1	65	25	35	82	34	34
2	48	9	8	68	7	11
3	20	5	2	1	0	1
Total documents	133	39	45	151	41	46
Total annotations	221	58	57	221	48	59

Table 3: Counts of numbers of papers with 1, 2 or 3 annotations in each section of each corpus.

all Multiply annotated versions of the same documents are not combined at all, but all versions are included in the training set. The machine learner is, in effect, used to reconcile the different versions on its own.

union Each set of annotations on the same document is combined by including all entities and relations marked by each annotator. For entity annotation there is a problem combining annotations in this way if entities cross (in other words, overlap but do not nest in the sense of Alex et al. (2007)), since crossing entities cannot be represented in the NER system. Crossing entities are resolved by removing the one with the later starting point.

intersection Annotations are combined by only choosing the entities and relations which are common to all annotated versions of a multiply annotated document.

one-random For each multiply-annotated document, one annotated version is chosen at random and the rest are discarded.

best-ner An NER model is trained using the **all** configuration. For each multiply annotated document in the training set, this NER model is then applied to all annotated versions in TRAIN and the version on which the model achieves the highest score is chosen. The rationale behind this method is that the system will do better on annotated versions which are more consistent with the rest of the corpus, so the combined corpus will be more consistent, and therefore provide better training data.

best-re The same as **best-ner**, except the annotated version is chosen by considering performance on relation extraction.

consistent An arbitrary order of preference is applied to the annotators, and for each multiply annotated document, the annotation corresponding to the annotator which came highest in the order of preference was selected. This combination method may create a more consistent corpus by favouring certain annotators.

Notice that **intersection** and **union** create a new version of the annotated document by combining annotations from each version, whereas **one-random**, **best-ner**, **best-re** and **consistent** simply choose one of the annotated versions, based on some criteria. We considered using a **voted** strategy, but since most of the multiply annotated documents only have two annotations, **voted** would just collapse to **intersection** or **union**, depending on how ties were resolved. Note that the variation in the training corpora produced by each combination strategy will be higher for relations than for entities, since the IAA (Table 4) is much lower for relations. The combined IAA score is relevant here, as it shows the overall difference between different annotators' views of the relations in the same document.

To compare the effect of adding extra multiply annotated data versus extra single annotated data, learning curves for each of these two configurations were produced. Firstly, all but one version of each multiply annotated document was removed from the original training set (DEVTEST and TRAIN combined), and a matching number of singly annotated documents was also removed. A model was then trained on the remaining training documents, and tested on TEST. Learning curves for singly and multiply annotated documents, respectively, were produced by re-adding the removed documents in 10 separate batches, training and testing after adding each batch. The whole process was repeated 20 times, averaging the results, with the documents placed in a different random order each time.

5 Results

The results for the different combination types on the NER task are shown in Table 5. Each combination type was tested in two different configurations; training on TRAIN and testing on DEVTEST, and training on TRAIN and DEVTEST combined then testing on TEST. The F_1 scores for each combination were micro-averaged across entity classes to give an overall F_1 score.

Method	PPI		TE	
	DEVTEST	TEST	DEVTEST	TEST
all	**75.1**	**72.5**	**65.4**	63.3
union	74.8	72.1	65.1	**63.7**
intersection	74.8	72.0	63.9	62.8
one-random	75.0	71.9	65.2	**63.7**
best-ner	74.7	72.1	64.9	63.5
best-re	**75.1**	72.1	65.1	63.6
consistent	74.7	72.4	65.0	63.6

Table 5: Comparison of performance of NER, training models on different versions of the training set produced by different combination methods. Performance is given on DEVTEST and TEST, and measured using F_1.

The differences in scores between each combination type appear small, and do not follow a consistent pattern. A Friedman rank sum test was performed separately on the PPI and TE TEST results, comparing the F_1 scores on each file, in each combination type. For both domains the test gave $p = 0.38$, indicating that there is no evidence of a significant difference between the methods.

The corresponding results for RE are show in Table 6, again comparing the performance of each combination method on DEVTEST and TEST, for both domains.

Method	PPI		TE	
	DEVTEST	TEST	DEVTEST	TEST
all	**58.9**	**58.7**	**58.3**	53.3
union	56.4	58.0	56.4	53.1
intersection	56.9	58.3	54.6	53.1
one-random	57.5	58.6	57.2	53.6
best-ner	57.9	58.5	57.0	**53.9**
best-re	57.7	58.6	57.0	52.8
consistent	57.8	58.6	57.0	53.4

Table 6: Comparison of performance of RE, training models on different versions of the training set produced by different combination methods. Performance is given on DEVTEST and TEST, and measured using F_1.

Testing with the Friedman rank sum test as for NER gives $p = 0.72$ for PPI and $p = 0.23$ for TE, again indicating that the effect of the different combination methods is not significant.

The effect of adding extra multiply annotated data to the training set, compared with adding extra singly annotated data, is shown in Figures 1 and 2. In each graph, the curve labelled "single" shows the performance of the system as more singly annotated data is added, and the "multiple" curve shows how performance changes as more multiply annotated data is added. The systems are trained on subsets of TRAIN and DEVTEST combined, and tested on TEST.

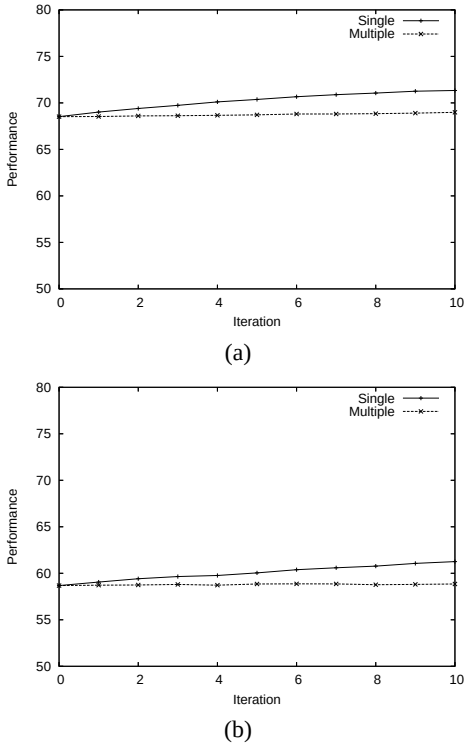


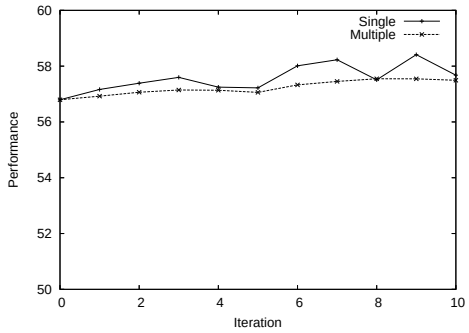
Figure 1: Comparison of the improvement gained from adding further singly annotated data, versus further multiply annotated data, for (a) PPI and (b) TE named entity recognition.

6 Discussion and Conclusion

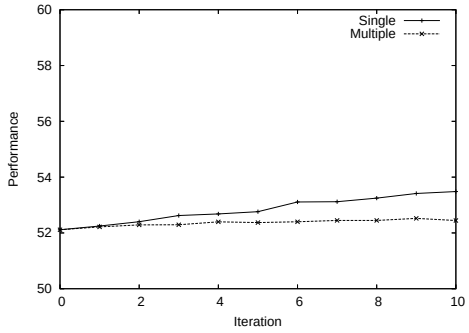
The comparisons between methods shown in Tables 5 and 6 indicate that there is very little difference between the different combination methods. For RE, there is a slight preference for the **all** combination method, but this preference is not significant, and could have arisen on the DEVTEST set because the system was optimised with the **all** combination method on that test set. The **union** and **intersection** combination strategies cause a change in the precision/recall balance (data not shown), since the former tends to increase the number of relations/entities in the annotated corpus, and the latter tends to reduce them.

It is perhaps surprising that using all the annotated versions of each document does not produce better performance than simply picking one of the versions at random. This is especially true for RE, where the IAA of below 60 F_1 means that there is a significant amount of extra information in the other annotated versions of a document. It is possible that because of the large training corpus, adding this extra data did not really offer anything extra, or that the inconsistency in the different versions of the annotated data just confused the machine learner. Maybe with a smaller training corpus the differences between different combination methods would be more marked.

The learning curves in Figure 2 suggest that adding extra multiple annotation can offer small improvements in NER



(a)



(b)

Figure 2: Comparison of the improvement gained from adding further singly annotated data, versus further multiply annotated data, for (a) PPI and (b) TE relation extraction.

and RE performance, but annotations of new documents will offer a faster rate of improvement. Whilst multiply annotated versions of documents may be required to measure IAA and to monitor annotation quality, it is better to keep the multiple annotation to the minimum required for these purposes, in order to maximise the budget available for extra singly annotated documents.

Acknowledgements

This work was carried out as part of an ITI Life Sciences Scotland (`http://www.itilifesciences.com`) research programme with Cognia EU (`http://www.cognia.com`) and the University of Edinburgh. The authors are grateful to Claire Grover and Ben Hachey for their comments.

7 References

Beatrice Alex, Malvina Nissim, and Claire Grover. 2006. The impact of annotation on the performance of protein tagging in biomedical text. In *Proceedings of LREC*, pages 595–600, Genoa, Italy.

Beatrice Alex, Barry Haddow, and Claire Grover. 2007. Recognising nested named entities in biomedical text. In *Proceedings of BioNLP*, pages 65–72, Prague, Czech Republic.

Beatrice Alex, Claire Grover, Barry Haddow, Mijail Kabadjov, Ewan Klein, Michael Matthews, Stuart Roebuck, Richard Tobin, and Xinglong Wang. 2008. Assisted curation: Does text mining really help? In Russ B. Altman, A. Keith Dunker, Lawrence Hunter, Tiffany Murray, and Teri E. Klein, editors, *Proceedings of the Pacific Symposium on Biocomputing 2008*, pages 556–567, Kohala Coast, Hawaii, USA.

Thorsten Brants. 2000. Inter-annotator agreement for a German newspaper corpus. In *Proceedings of LREC*, pages 1435–1439, Athens, Greece.

James Curran and Stephen Clark. 2003. Language independent NER using a maximum entropy tagger. In *Proceedings of CoNLL*, pages 164–167, Edmonton, Canada.

Barry Haddow and Michael Matthews. 2007. The extraction of enriched protein-protein interactions from biomedical text. In *Proceedings of BioNLP*, pages 145–152, Prague, Czech Republic.

Mani Inderjeet, Hu Zhangzhi, Jang S. Bae, Samuel Ken, Krause Matthew, Phillips Jon, and Wu. 2005. Protein name tagging guidelines: lessons learned. *Comparative and Functional Genomics*, 6(1-2):72–76.

Zhiyong Lu, Michael Bada, Philip V. Ogren, K. Bretonnel Cohen, and Lawrence Hunter. 2006. Improving biomedical corpus annotation guidelines. In *Proceedings of the Joint BioLINK and 9th Bio-Ontologies Meeting*, pages 89–92, Fortaleza, Brasil.

Tara McIntosh and James Curran. 2007. Challenges for extracting biomedical knowledge from full text. In *Proceedings of BioNLP*, pages 171–178, Prague, Czech Republic.

Leif Arda Nielsen. 2006. Extracting protein-protein interactions using simple contextual features. In *Proceedings of BioNLP*, pages 120–121, New York, USA.

Parantu K. Shah, Carolina Perez-Iratxeta, Peer Bork, and Miguel A. Andrade. 2003. Information extraction from full text scientific articles: where are the keywords? *BMC Bioinformatics*, 4(20).

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of CoNLL*, pages 142–147, Edmonton, Canada.