Assessing the Practical Usability of an Automatically Annotated Corpus

Md. Faisal Mahbub Chowdhury [†‡] **and Alberto Lavelli** [‡]
[‡] Human Language Technology Research Unit, Fondazione Bruno Kessler, Trento, Italy
[†] Department of Information Engineering and Computer Science, University of Trento, Italy
{chowdhury,lavelli}@fbk.eu

Abstract

The creation of a gold standard corpus (GSC) is a very laborious and costly process. Silver standard corpus (SSC) annotation is a very recent direction of corpus development which relies on multiple systems instead of human annotators. In this paper, we investigate the practical usability of an SSC when a machine learning system is trained on it and tested on an unseen benchmark GSC. The main focus of this paper is how an SSC can be maximally exploited. In this process, we inspect several hypotheses which might have influenced the idea of SSC creation. Empirical results suggest that some of the hypotheses (e.g. a positive impact of a large SSC despite of having wrong and missing annotations) are not fully correct. We show that it is possible to automatically improve the quality and the quantity of the SSC annotations. We also observe that considering only those sentences of SSC which contain annotations rather than the full SSC results in a performance boost.

1 Introduction

The creation of a **gold standard corpus (GSC)** is not only a very laborious task due to the manual effort involved but also a costly and time consuming process. However, the importance of the GSC to effectively train machine learning (ML) systems cannot be underestimated. Researchers have been trying for years to find alternatives or at least some compromise. As a result, self-training, co-training and unsupervised approaches targeted for specific tasks (such as word sense disambiguation, syntactic parsing, etc) have emerged. In the process of these researches, it became clear that the size of the (manually annotated) training corpus has an impact on the final outcome.

Recently an initiative is ongoing in the context of the European project CALBC[1] which aims to create a large, so called **silver standard corpus (SSC)** using harmonized annotations automatically produced by multiple systems (Rebholz-Schuhmann et al., 2010; Rebholz-Schuhmann et al., 2010a; Rebholz-Schuhmann et al., 2010b). The basic idea is that independent biomedical named entity recognition (BNER) systems annotate a large corpus of biomedical articles without any restriction on the methodology or external resources to be exploited. The different annotations are automatically harmonized using some criteria (e.g. minimum number of systems to agree on a certain annotation) to yield a consensus based corpus. This consensus based corpus is called silver standard corpus because, differently from a GSC, it is not created exclusively by human annotators. Several factors can influence the quantity and quality of the annotations during SSC development. These include varying performance, methodology, annotation guidelines and resources of the SSC annotation systems (henceforth **annotation systems**).

The annotation of SSC in the framework of the CALBC project is focused on (bio) entity mentions (a specific application of the named entity recognition (NER)[2] task). However, the idea of SSC creation might also be applied to other types of annotations, e.g. annotation of relations among entities, annotation of treebanks and so on. Hence, if it can be

[1] http://www.ebi.ac.uk/Rebholz-srv/CALBC/project.html

[2] Named entity recognition is the task of locating boundaries of the entity mentions in a text and tagging them with their corresponding semantic types (e.g. person, location, disease and so on).

Proceedings of the Fifth Law Workshop (LAW V), pages 101–109,
Portland, Oregon, 23-24 June 2011.

shown that an SSC is a useful resource for the NER task, similar resources can be developed for annotation of information other than entities and utilized for the relevant natural language processing (NLP) tasks.

The primary objective of SSC annotation is to compensate the cost, time and manual effort required for a GSC. The procedure of SSC development is inexpensive, fast and yet capable of yielding huge amount of annotated data. These advantages trigger several hypotheses. For example:

- The size of annotated training corpus always plays a crucial role in the performance of ML systems. If the annotation systems have very high precision and somewhat moderate recall, they would be also able to annotate automatically a huge SSC which would have a good quality of annotations. So, one might assume that, even if such an SSC may contain wrong and missing annotations, a relatively 15 or 20 times bigger SSC than a smaller GSC should allow an ML based system to ameliorate the adverse effects of the erroneous annotations.

- Rebholz-Schuhmann et al. (2010) hypothesized that an SSC might serve as an approximation of a GSC.

- In the absence of a GSC, it is expected that ML systems would be able to exploit the harmonised annotations of an SSC to annotate unseen text with reasonable accuracy.

- An SSC could be used to semi-automate the annotations of a GSC. However, in that case, it is expected that the annotation systems would have very high recall. One can assume that converting an SSC into a GSC would be less time consuming and less costly than developing a GSC from scratch.

All these hypotheses are yet to be verified. Nevertheless, once we have an SSC annotated with certain type of information, the main question would be *how this corpus can be maximally exploited* given the fact that it might be created by annotation systems that used different resources and possibly not the same annotation guidelines. This question is directly related to the practical usability of an SSC, which is the focus of this paper.

Taking the aforementioned hypotheses into account, our goal is to investigate the following research questions which are fundamental to the maximum exploitation of an SSC:

1. How can the annotation quality of an SSC be improved automatically?

2. How would a system trained on an SSC perform if tested on an unseen benchmark GSC?

3. Can an SSC combined with a GSC produce a better trained system?

4. What would be the impact on system performance if *unannotated sentences*[3] are removed from an SSC?

5. What would be the effects of the variation in the size of an SSC on precision and recall?

Our goal is not to judge the procedure of SSC creation, rather our objective is to examine how an SSC can be exploited *automatically* and *maximally* for a specific task. Perhaps this would provide useful insights to re-evaluate the approach of SSC creation.

For our experiments, we use a benchmark GSC called the BioCreAtIvE II GM corpus (Smith et al., 2008) and the CALBC SSC-I corpus (Rebholz-Schuhmann et al., 2010a). Both of these corpora are annotated with genes. Our motivation behind the choice of a gene annotated GSC for the SSC evaluation is that ML based BNER for genes has already achieved a sufficient level of maturity. This is not the case for other important bio-entity types, primarily due to the absence of training GSC of adequate size. In fact, for many bio-entity types there exist no GSC. If we can achieve a reasonably good baseline for gene mention identification by maximizing the exploitation of SSC, we might be able to apply almost similar strategies to exploit SSC for other bio-entity types, too.

The remaining of this paper is organised as follows. Section 2 includes brief discussion of the related work. Apart from mentioning the related literature, this section also underlines the difference of

[3]For the specific SSC that we use in this work, *unannotated sentences* correspond to those sentences that contain no gene annotation.

SSC development with respect to approaches such as self-training and co-training. Then in Section 3, we describe the data used in our experiments and the experimental settings. Following that, in Section 4, empirical results are presented and discussed. Finally, we conclude with a description of what we learned from this work in Section 5.

2 Related Work

As mentioned, the concept of SSC has been initiated by the CALBC project (Rebholz-Schuhmann et al., 2010a; Rebholz-Schuhmann et al., 2010). So far, two versions of SSC have been released as part of the project. The CALBC SSC-I has been harmonised from the annotations of the systems provided by the four project partners. Three of them are dictionary based systems while the other is an ML based system. The systems utilized different types of resources such as GENIA corpus (Kim et al., 2003), Entrez Genes[4], Uniprot[5], etc. The CALBC SSC-II corpus has been harmonised from the annotations done by the 11 participants of the first CALBC challenge and the project partners.[6] Some of the participants have used the CALBC SSC-I versions for training while others used various gene databases or benchmark GSCs such as the BioCreAtIvE II GM corpus.

One of the key questions regarding an SSC would be how close its annotation quality is to a corresponding GSC. On the one hand, every GSC contains its special view of the correct annotation of a given corpus. On the other hand, an SSC is created by systems that might be trained with resources having different annotation standards. So, it is possible that the annotations of an SSC significantly differ with respect to a manually annotated (i.e., gold standard) version of the same corpus. This is because human experts are asked to follow specific annotation guidelines.

Rebholz-Schuhmann and Hahn (2010c) did an intrinsic evaluation of the SSC where they created an SSC and a GSC on a dataset of 3,236 Medline[7] abstracts. They were not able to make any specific conclusion whether the SSC is approaching to the GSC. They were of the opinion that SSC annotations are more similar to terminological resources.

Hahn et al. (2010) proposed a policy where silver standards can be dynamically optimized and customized on demand (given a specific goal function) using a gold standard as an oracle. The gold standard is used for optimization only, not for training for the purpose of SSC annotation. They argued that the nature of diverging tasks to be solved, the levels of specificity to be reached, the sort of guidelines being preferred, etc should allow prospective users of an SSC to customize one on their own and not stick to something that is already prefabricated without concrete application in mind.

Self-training and co-training are two of the existing approaches that have been used for compensating the lack of a training GSC with adequate size in several different tasks such as word sense disambiguation, semantic role labelling, parsing, etc (Ng and Cardie, 2003; Pierce and Cardie, 2004; McClosky et al., 2006; He and Gildea, 2006). According to Ng and Cardie (2003), self-training is the procedure where a committee of classifiers are trained on the (gold) annotated examples to tag unannotated examples independently. Only those new annotations to which all the classifiers agree are added to the training set and classifiers are retrained. This procedure repeats until a stop condition is met. According to Clark et al. (2003), self-training is a procedure in which "a tagger is retrained on its own labeled cache at each round". In other words, a single classifier is trained on the initially (gold) annotated data and then applied on a set of unannotated data. Those examples meeting a selection criterion are added to the annotated dataset and the classifier is retrained on this new data set. This procedure can continue for several rounds as required.

Co-training is another weakly supervised approach (Blum and Mitchell, 1998). It applies for those tasks where each of the two (or more) sets of features from the initially (gold) annotated training data is sufficient to classify/annotate the unannotated data (Pierce and Cardie, 2001; Pierce and Cardie,

[4] http://jura.wi.mit.edu/entrez_gene/

[5] http://www.uniprot.org/

[6] See proceedings of the 1st CALBC Workshop, 2010, Editors: Dietrich Rebholz-Schuhmann and Udo Hahn (http://www.ebi.ac.uk/Rebholz-srv/CALBC/docs/FirstProceedings.pdf) for details.

[7] http://www.nlm.nih.gov/databases/databases_medline.html

2004; He and Gildea, 2006). As with SSC annotation and self-training, it also attempts to increase the amount of annotated data by making use of unannotated data. The main idea of co-training is to represent the initially annotated data using two (or more) separate feature sets, each called a "view". Then, two (or more) classifiers are trained on those views of the data which are then used to tag new unannotated data. From this newly annotated data, the most confident predictions are added to the previously annotated data. This whole process may continue for several iterations. It should be noted that, by limiting the number of views to one, co-training becomes self-training.

Like the SSC, the multiple classifier approach of self-training and co-training, as described above, adopts the same vision of utilizing automatic systems for producing the annotation. Apart from that, SSC annotation is completely different from both self-training and co-training. For example, classifiers in self-training and co-training utilizes the same (manually annotated) resource for their initial training. But SSC annotation systems do not necessarily use the same resource. Both self-training and co-training are weakly supervised approaches where the classifiers are based on supervised ML techniques. In the case of SSC annotation, the annotation systems can be dictionary based or rule based. This attractive flexibility allows SSC annotation to be a completely unsupervised approach since the annotation systems do not necessarily need to be trained.

3 Experimental settings

We use the BioCreAtIvE II GM corpus (henceforth, only the GSC) for evaluation of an SSC. The training corpus in the GSC has in total 18,265 gene annotations in 15,000 sentences. The GSC test data has 6,331 annotations in 5,000 sentences.

Some of the CALBC challenge participants have used the BioCreAtIvE II GM corpus for training to annotate gene/protein in the CALBC SSC-II corpus. We wanted our benchmark corpus and benchmark corpus annotation to be totally unseen by the systems that annotated the SSC to be used in our experiments so that there is no bias in our empirical results. SSC-I satisfies this criteria. So, we use the SSC-I (henceforth, we would refer the CALBC SSC-I as simply the SSC) in our experiments despite the fact that it is almost 3 times smaller than the SSC-II. The SSC has in total 137,610 gene annotations in 316,869 sentences of 50,000 abstracts.

Generally, using a customized dictionary of entity names along with annotated corpus boosts NER performance. However, since our objective is to observe to what extent a ML system can learn from SSC, we avoid the use of any dictionary. We use an open source ML based BNER system named BioEnEx[8] (Chowdhury and Lavelli, 2010). The system uses conditional random fields (CRFs), and achieves comparable results (F_1 score of 86.22% on the BioCreAtIvE II GM test corpus) to that of the other state-of-the-art systems without using any dictionary or lexicon.

One of the complex issues in NER is to come to an agreement regarding the boundaries of entity mentions. Different annotation guidelines have different preferences. There may be tasks where a longer entity mention such as "human IL-7 protein" may be appropriate, while for another task a short one such as "IL-7" is adequate (Hahn et al., 2010).

However, usually evaluation on BNER corpora (e.g., the BioCreAtIvE II GM corpus) is performed adopting exact boundary match. Given that we have used the official evaluation script of the BioCreAtIvE II GM corpus, we have been forced to adopt exact boundary match. Considering a relaxed boundary matching (i.e. the annotations might differ in uninformative terms such as *the*, *a*, *acute*, etc.) rather than exact boundary matching might provide a slightly different picture of the effectiveness of the SSC usage.

4 Results and analyses

4.1 Automatically improving SSC quality

The CALBC SSC-I corpus has a negligible number of overlapping gene annotations (in fact, only 6). For those overlapping annotations, we kept only the longest ones. Our hypothesis is that a certain token in the same context can refer to (or be part of) only one concept name (i.e. annotation) of a certain semantic group (i.e. entity type). After removing these few overlaps, the SSC has 137,604 annotations. We

[8] Freely available at http://hlt.fbk.eu/en/people/chowdhury/research

will refer to this version of the SSC as the **initial SSC (ISSC)**.

We construct a list[9] using the lemmatized form of 132 frequently used words that appear in gene names. These words cannot constitute a gene name themselves. If (the lemmatized form of) all the words in a gene name belong to this list then that gene annotation should be discarded. We use this list to remove erroneous annotations in the ISSC. After this purification step, the total number of annotations is reduced to 133,707. We would refer to this version as the **filtered SSC (FSSC)**.

Then, we use the post-processing module of BioEnEx, first to further filter out possible wrong gene annotations in the FSSC and then to automatically include potential gene mentions which are not annotated. It has been observed that some of the annotated mentions in the SSC-I span only part of the corresponding token[10]. For example, in the token "IL-2R", only "IL-" is annotated. We extend the post-processing module of BioEnEx to automatically identify all such types of annotations and expand their boundaries when their neighbouring characters are alphanumeric.

Following that, the extended post-processing module of BioEnEx is used to check in every sentence whether there exist any potential unannotated mentions[11] which differ from any of the annotated mentions (in the same sentence) by a single character (e.g. "IL-2L" and "IL-2R"), number (e.g. "IL-2R" and "IL-345R") or Greek letter (e.g. "IFN-alpha" and "IFN-beta"). After this step, the total number of gene annotations is 144,375. This means that *we were able to remove/correct some specific types of errors and then further expand the total number of annotations (by including entities not annotated in the original SSC) up to **4.92%** with respect to the ISSC*. We will refer to this expanded version of the SSC as the **processed SSC (PSSC)**.

When BioEnEx is trained on the above versions of the SSC and tested on the GSC test data, we observed an increase of more than 3% of F_1 score because of the filtering and expansion (see Table 1). One noticeable characteristic in the results is that the number of annotations obtained (i.e. TP+FP[12]) by training on any of the versions of the SSC is almost half of the actual number annotations of the GSC test data. This has resulted in a low recall. There could be mainly two reasons behind this outcome:

	TP	**FP**	**FN**	**P**	**R**	F_1
ISSC	2,396	594	3,935	80.13	37.85	51.41
FSSC	2,518	557	3,813	81.89	39.77	53.54
PSSC	2,606	631	3,725	80.51	41.16	54.47

Table 1: The results of experiments when trained with different versions of the SSC and tested on the GSC test data.

- First of all, it might be the case that a considerable number of gene names are not annotated inside the SSC versions. As a result, the features shared by the annotated gene names (i.e. TP) and unnannotated gene names (i.e. FN) might not have enough influence.

- There might be a considerable number of wrong annotations which are actually not genes (i.e. FP). Consequently, a number of bad features might be collected from those wrong annotations which are misleading the training process.

To verify the above conditions, it would be required to annotate the huge CALBC SSC manually. This would be not feasible because of the cost of human labour and time. Nevertheless, we can try to measure the state of the above conditions roughly by using only *annotated sentences* (i.e. sentences containing at least one annotation) and varying the size of the corpus, which are the subjects of our next experiments.

[9]The words are collected from http://pir.georgetown.edu/pirwww/iprolink/general_name and the annotation guideline of GENETAG (Tanabe et al., 2005).

[10]By *token* we mean a sequence of consecutive non-whitespace characters.

[11]Any token or sequence of tokens is considered to verify whether it should be annotated or not, if its length is more than 2 characters excluding digits and Greek letters.

[12]TP (true positive) = corresponding annotation done by the system is correct, FP (false positive) = corresponding annotation done by the system is incorrect, FN (false negative) = corresponding annotation is correct but it is not annotated by the system.

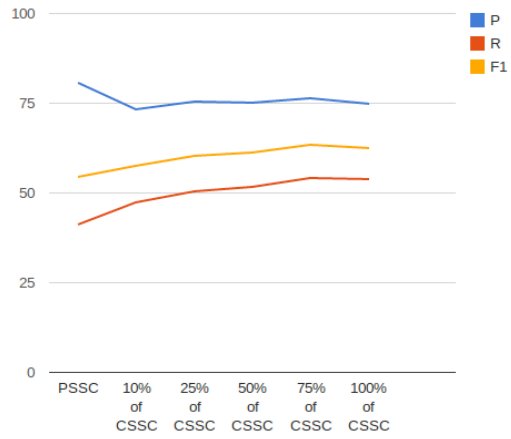

Figure 1: Graphical representation of the experimental results with varying size of the CSSC.

4.2 Impact of annotated sentences and different sizes of the SSC

We observe that only 77,117 out of the 316,869 sentences in the PSSC contain gene annotations. We will refer to the sentences having at least one gene annotation collectively as the **condensed SSC (CSSC)**. Table 2 and Figure 1 show the results when we used different portions of the CSSC for training.

There are four immediate observations on the above results:

- Using the full PSSC, we obtain total (i.e. TP+FP) 3,237 annotations on the GSC test data. But when we use only annotated sentences of the PSSC (i.e. the CSSC), the total number of annotations is 4,562, i.e. there is an increment of 40.93%.
- Although we have a boost in F_1 score due to the increase in recall using the CSSC in place of the PSSC, there is a considerable drop in precision.
- The number of FP is almost the same for the usage of 10-75% of the CSSC.
- The number of FN kept decreasing (and TP kept increasing) for 10-75% of the CSSC.

These observations can be interpreted as follows:

- Unannotated sentences inside the SSC in reality contain many gene annotations; so the inclusion of such sentences misleads the training process of the ML system.
- Some of the unannotated sentences actually do not contain any gene names, while others would contain such names but the automatic annotations missed them. As a consequence, the former sentences contain true negative examples which could provide useful features that can be exploited during training so that less FPs are produced (with a precision drop using the CSSC). So, instead of simply discarding all the unannotated sentences, we could adopt a filtering strategy that tries to distinguish between the two classes of sentences above.
- The experimental results with the increasing size of the CSSC show a decrease in both precision (74.55 vs 76.17) and recall (53.72 vs 54.04). We plan to run again these experiments with different randomized splits to better assess the performance.
- Even using only 10% of the whole CSSC does not produce a drastic difference with the results when the full CSSC is used. This indicates that perhaps the more CSSC data is fed, the more the system tends to overfit.
- It is evident that the more the size of the CSSC increases, the lower the improvement of F_1 score, if the total number of annotations in the newly added sentences and the accuracy of the annotations are not considerably higher. It might be not surprising if, after the addition of more sentences in the CSSC, the F_1 score drops further rather than increasing. The assumption that having a huge SSC would be beneficiary might not be completely correct. There might be some optimal limit of the SSC (depending on the task) that can provide maximum benefits.

4.3 Training with the GSC and the SSC together

Our final experiments were focused on whether it is possible to improve performance by simply merging the GSC training data with the PSSC and the CSSC. The PSSC has almost 24 times the number of sentences and almost 8 times the number of gene annotations than the GSC. There is a possibility that, when we do a simple merge, the weight of the

	Total tokens in the corpus	No of annotated genes	TP	FP	FN	P	R	F_1
PSSC	6,955,662	144,375	2,606	631	3,725	80.51	41.16	54.47
100% of CSSC	1,983,113	144,375	3,401	1,161	2,930	74.55	53.72	62.44
75% of CSSC	1,487,823	108,213	3,421	1,070	2,910	76.17	54.04	63.22
50% of CSSC	992,392	72,316	3,265	1,095	3,066	74.89	51.57	61.08
25% of CSSC	494,249	35,984	3,179	1,048	3,152	75.21	50.21	60.22
10% of CSSC	196,522	14,189	2,988	1,097	3,343	73.15	47.20	57.37

Table 2: The results of SSC experiments with varying size of the CSSC = condensed SSC (i.e. sentences containing at least one annotation). SSC size = 316,869 sentences. CSSC size = 77,117.

	TP	FP	FN	P	R	F_1
GSC	5,373	759	958	87.62	84.87	86.22
PSSC + GSC	3,745	634	2,586	85.52	59.15	69.93
PSSC + GSC * 8	4,163	606	2,168	87.29	65.76	75.01
CSSC + GSC * 8	4,507	814	1,824	84.70	71.19	77.36

Table 3: The results of experiments by training on the GSC training data merged with the PSSC and the CSSC.

gold annotations would be underestimated. So, apart from doing a simple merge, we also try to balance the annotations of the two corpora. There are two options to do this – (i) by duplicating the GSC training corpus 8 times to make its total number of annotations equal to that of the PSSC, or (ii) by choosing randomly a portion of the PSSC that would have almost similar amount of annotations as that of the GSC. We choose the 1st option.

Unfortunately, when an SSC (i.e. the PSSC or the CSSC) is combined with the GSC, the result is far below than that of using the GSC only (see Table 3). Again, low recall is the main issue partly due to the lower number of annotations (i.e. TP+FP) done by the system trained on an SSC and the GSC instead of the GSC only. As we know, a GSC is manually annotated following precise guidelines, while an SSC is annotated with automatic systems that do not necessarily follow the same guidelines as a GSC. So, it would not have been surprising if the number of annotations were high (since we have much bigger training corpus due to SSC) but precision were low. But in practice, precision obtained by combining an SSC and the GSC is almost as high as the precision achieved using the GSC.

One reason for the lower number of annotations might be the errors that have been propagated inside the SSC. Some of the systems that have been used for the annotation of the SSC might have low recall. As a result, during harmonization of their annotations several valid gene mentions might not have been included[13].

One other possible reason could be the difference in the entity name boundaries in the GSC and an SSC. We have checked some of the SSC annotations randomly. It appears that in those annotated entity names some relevant (neighbouring) words (in the corresponding sentences) are not included. It is most likely that the SSC annotation systems had disagreements on those words.

When the annotations of the GSC were given higher preference (by duplicating), there is a substantial improvement in the F_1 score, although still lower than the result with the GSC only.

5 Conclusions

The idea of SSC development is simple and yet attractive. Obtaining better results on a test dataset by combining output of multiple (accurate and diverse[14]) systems is not new (Torii et al., 2009; Smith et al., 2008). But adopting this strategy for cor-

[13]There can be two reasons for this – (i) when a certain valid gene name is not annotated by any of the annotation systems, and (ii) when only a few of those systems have annotated the valid name but the total number of such systems is below than the minimum required number of agreements, and hence the gene name is not considered as an SSC annotation.

[14]A system is said to be accurate if its classification performance is better than a random classification. Two systems are considered diverse if they do not make the same classification mistakes. (Torii et al., 2009)

pus development is a novel and unconventional approach. Some natural language processing tasks (especially the new ones) lack adequate GSCs to be used for the training of ML based systems. For such tasks, domain experts can provide patterns or rules to build systems that can be used to annotate an initial version of SSC. Such systems might lack high recall but are expected to have high precision. Already available task specific lexicons or dictionaries can also be utilized for SSC annotation. Such an initial version of SSC can be later enriched using automatic process which would utilize existing annotations in the SSC.

With this vision in mind, we pose ourselves several questions (see Section 1) regarding the practical usability and exploitation of an SSC. Our experiments are conducted on a publicly available biomedical SSC developed for the training of biomedical NER systems. For the evaluation of a state-of-the-art ML system trained on such an SSC, we use a widely used benchmark biomedical GSC.

In the search of answers for our questions, we accumulate several important empirical observations. We have been able to automatically reduce the number of erroneous annotations from the SSC and include unannotated potential entity mentions simply using the annotations that the SSC already provides. Our techniques have been effective for improving the annotation quality as there is a considerable increment of F_1 score (almost 11% higher when we use CSSC instead of using ISSC; see Table 1 and 2).

We also observe that it is possible to obtain more than 80% of precision using the SSC. But recall remains quite low, partly due to the low number of annotations provided by the system trained with the SSC. Perhaps, the entity names in the SSC that are missed by the annotation systems is one of the reasons for that.

Perhaps, the most interesting outcome of this study is that, if only annotated sentences (which we call *condensed* corpus) are considered, then the number of annotations as well as the performance increases significantly. This indicates that many unannotated sentences contain annotations missed by the automatic annotation systems. However, it appears that correctly unannotated sentences influence the achievement of high precision. Maybe a more sophisticated approach should be adopted instead of completely discarding the unannotated sentences, e.g. devising a filter able to distinguish between relevant unannotated sentences (i.e., those that should contain annotations) from non-relevant ones (i.e., those that correctly do not contain any annotation). Measuring lexical similarity between annotated and unannotated sentences might help in this case.

We notice the size of an SSC affects performance, but increasing it above a certain limit does not always guarantee an improvement of performance (see Figure 1). This rejects the hypothesis that having a much larger SSC should allow an ML based system to ameliorate the effect of having erroneous annotations inside the SSC.

Our empirical results show that combining GSC and SSC do not improve results for the particular task of NER, even if GSC annotations are given higher weights (through duplication). We assume that this is partly due to the variations in the guidelines of entity name boundaries[15]. These impact the learning of the ML algorithm. For other NLP tasks where the possible outcome is boolean (e.g. relation extraction, i.e. whether a particular relation holds between two entities or not), we speculate the results of such combination might be better.

We use a CRF based ML system for our experiments. It would be interesting to see whether the observations are similar if a system with a different ML algorithm is used.

To conclude, this study suggests that an automatically pre-processed SSC might already contain annotations with reasonable quality and quantity, since using it we are able to reach more than 62% of F_1 score. This is encouraging since in the absence of a GSC, an ML system would be able to exploit an SSC to annotate unseen text with a moderate (if not high) accuracy. Hence, SSC development might be a good option to semi-automate the annotation of a GSC.

Acknowledgments

This work was carried out in the context of the project "eOnco - Pervasive knowledge and data management in cancer care". The authors would like to thank Pierre Zweigenbaum for useful discussion, and the anonymous reviewers for valuable feedback.

[15]For example, "human IL-7 protein" vs "IL-7".

References

Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational learning theory (COLT'98)*, pages 92–100.

Md. Faisal Mahbub Chowdhury and Alberto Lavelli. 2010. Disease mention recognition with specific features. In *Proceedings of the Workshop on Biomedical Natural Language Processing (BioNLP 2010), 48th Annual Meeting of the Association for Computational Linguistics*, pages 83–90, Uppsala, Sweden, July.

Stephen Clark, James R. Curran, and Miles Osborne. 2003. Bootstrapping POS taggers using unlabelled data. In *Proceedings of the 7th Conference on Natural Language Learning (CoNLL-2003)*, pages 49–55.

Udo Hahn, Katrin Tomanek, Elena Beisswanger, and Erik Faessler. 2010. A proposal for a configurable silver standard. In *Proceedings of the 4th Linguistic Annotation Workshop, 48th Annual Meeting of the Association for Computational Linguistics*, pages 235–242, Uppsala, Sweden, July.

Shan He and Daniel Gildea. 2006. Self-training and co-training for semantic role labeling: Primary report. Technical report, University of Rochester.

Jin-Dong Kim, Tomoko Ohta, Yuka Tateisi, and Jun'ichi Tsujii. 2003. Genia corpus - semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(Suppl 1):i180–182.

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Reranking and self-training for parser adaptation. In *Proceedings of the 21st International Conference on Computational Linguistics*, pages 337–344, Sydney, Australia.

Vincent Ng and Claire Cardie. 2003. Weakly supervised natural language learning without redundant views. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL-2003)*, pages 173–180.

David Pierce and Claire Cardie. 2001. Limitations of co-training for natural language learning from large datasets. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP-2001)*, pages 1–9.

David Pierce and Claire Cardie. 2004. Co-training and self-training for word sense disambiguation. In *Proceedings of the 8th Conference on Computational Natural Language Learning (CoNLL-2004)*, pages 33–40.

Dietrich Rebholz-Schuhmann and Udo Hahn. 2010c. Silver standard corpus vs. gold standard corpus. In *Proceedings of the 1st CALBC Workshop*, Cambridge, U.K., June.

Dietrich Rebholz-Schuhmann, Antonio Jimeno, Chen Li, Senay Kafkas, Ian Lewin, Ning Kang, Peter Corbett, David Milward, Ekaterina Buyko, Elena Beisswanger, Kerstin Hornbostel, Alexandre Kouznetsov, Rene Witte, Jonas B Laurila, Christopher JO Baker, Chen-Ju Kuo, Simon Clematide, Fabio Rinaldi, Richrd Farkas, Gyrgy Mra, Kazuo Hara, Laura Furlong, Michael Rautschka, Mariana Lara Neves, Alberto Pascual-Montano, Qi Wei, Nigel Collier, Md. Faisal Mahbub Chowdhury, Alberto Lavelli, Rafael Berlanga, Roser Morante, Vincent Van Asch, Walter Daelemans, José Luís Marina, Erik van Mulligen, Jan Kors, and Udo Hahn. 2010. Assessment of NER solutions against the first and second CALBC silver standard corpus. In *Proceedings of the fourth International Symposium on Semantic Mining in Biomedicine (SMBM'2010)*, October.

Dietrich Rebholz-Schuhmann, Antonio José Jimeno-Yepes, Erik van Mulligen, Ning Kang, Jan Kors, David Milward, Peter Corbett, Ekaterina Buyko, Elena Beisswanger, and Udo Hahn. 2010a. CALBC silver standard corpus. *Journal of Bioinformatics and Computational Biology*, 8:163–179.

Dietrich Rebholz-Schuhmann, Antonio José Jimeno-Yepes, Erik van Mulligen, Ning Kang, Jan Kors, David Milward, Peter Corbett, Ekaterina Buyko, Katrin Tomanek, Elena Beisswanger, and Udo Hahn. 2010b. The CALBC silver standard corpus for biomedical named entities – a study in harmonizing the contributions from four independent named entity taggers. In *Proceedings of the 7th International conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May.

Larry Smith, Lorraine Tanabe, Rie Ando, Cheng-Ju Kuo, I-Fang Chung, Chun-Nan Hsu, Yu-Shi Lin, Roman Klinger, Christoph Friedrich, Kuzman Ganchev, Manabu Torii, Hongfang Liu, Barry Haddow, Craig Struble, Richard Povinelli, Andreas Vlachos, William Baumgartner, Lawrence Hunter, Bob Carpenter, Richard Tsai, Hong-Jie Dai, Feng Liu, Yifei Chen, Chengjie Sun, Sophia Katrenko, Pieter Adriaans, Christian Blaschke, Rafael Torres, Mariana Neves, Preslav Nakov, Anna Divoli, Manuel Mana-Lopez, Jacinto Mata, and W John Wilbur. 2008. Overview of BioCreAtIvE II gene mention recognition. *Genome Biology*, 9(Suppl 2):S2.

Lorraine Tanabe, Natalie Xie, Lynne Thom, Wayne Matten, and W John Wilbur. 2005. GENETAG: a tagged corpus for gene/protein named entity recognition. *BMC Bioinformatics*, 6(Suppl 1):S3.

Manabu Torii, Zhangzhi Hu, Cathy H Wu, and Hongfang Liu. 2009. Biotagger-GM: a gene/protein name recognition system. *Journal of the American Medical Informatics Association : JAMIA*, 16:247–255.