Concept, Assertion and Relation Extraction at the 2010 i2b2 Relation Extraction Challenge using parsing information and dictionaries

Illés Solt[1], Ferenc P. Szidarovszky[1], Domonkos Tikk, PhD[1,2]

[1] Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary

[2] Department of Computer Science, Humboldt-University of Berlin, D-12489 Berlin, Rudower Chaussee 25, Germany

{illes.solt,tikk,szidarovszky}@tmit.bme.hu

Abstract

Medical discharge summaries are rich source of information for physicians, but to allow the further, e.g. statistical, analysis the text should be preprocessed. The yearly organized i2b2 shared tasks provide an arena where different text extraction techniques could be compared on given problems. In the Relation Extraction Challenge a three-level annotation task was set for discharge summaries: (1) concepts that may contribute to the focused relation, (2) the assertive attribute of these concepts, (3) and finally the relation between the concepts had to be identified. Here we present our solution for all three tasks. For concept identification and assertion classification, we applied dictionary and rule based solutions working on the syntactic parse trees of sentences. For relation extraction, we applied both machine learning approaches (support vector machines with various kernel functions) and a simple co-occurrence based method. On the test data sets, our methods achieved F-measures of 57% (exact) or 83% (inexact) in concept identification, 92% in assertion classification and 68% in relation extraction.

Introduction

Biomedical text mining has become a thriving field recently, because it proved its efficiency in different application areas, such as e.g. the identification of biological entities (proteins, genes, etc.) in free text [1], assigning insurance codes to clinical records [2], facilitating querying in biomedical databases [3], discovering connections between genes and diseases [4] – for a survey see [5].

Clinical texts like discharge summaries offer a rich source of information for the information extraction (IE) tasks. By using IE methods statistically relevant data can be extracted from these texts, which may help physicians in making medical studies.

Problem definition

The i2b2 Relation Extraction Challenge consists of 3 tasks.[1] The concept extraction task is restricted to the identification of the following concept types: medical problems, treatments and tests. Within these concept types further limitations apply, depending e.g. on the location (e.g. heading) or the grammatical structure of the occurrence of the concept. In this regards only complete noun phrases (NP) or adverbial phrases (AP) had to be marked.

In the assertion classification task, concepts extracted in task 1 have to be classified into one of the following six assertion categories: present, absent, possible, conditional, hypothetical, not related to the patient. The assertion class has to be determined based on the context of the concept in the clinical text.

In the relation extraction task, the type of binary relationship between concepts (if any) has to be determined. There are 8 predefined relationship types between given types of concepts. All but one are symmetric relationships, only the *problem indicates problem* (PIP) type is asymmetric; here the direction of the indication is essential.

The challenge was organized in three steps. After the completion of the task 1, its ground truth was given to participants to perform task 2; similarly after task 2 its ground truth was published. This evaluation process facilitates the separate evaluation of methods created for each task, since the error is not additive. However, one should obviously count with larger errors at assertion classification and relation

[1] A detailed description of the tasks can be found at https://www.i2b2.org/NLP/Relations/Documentation.php

extraction when the three tasks are performed in a sequel. To quantify the increase of error e.g. for relation extraction, see [6], where the combination of entity recognition and protein–protein interaction (PPI) relation extraction is investigated.

Methods

Concept identification

The concept annotation task can be formulated as a multi-label token sequence tagging problem, where the labels correspond to the three concept classes: problems, treatments and tests. According to the Annotation Guidelines (AG), annotations should meet certain linguistic requirements, e.g., they should include the head of a noun or adjective phrase. Our approach was to identify head terms and their classes (termed matching); and for each head term, find the containing span which also fulfills the AG requirements (termed expansion).

To identify head terms we used dictionaries built for the past i2b2 Obesity Challenge, as well as terms extracted from the training data. A term in the training data was considered for its class if its inexact span instance precision (occurrences overlapping with annotations of the correct type/all occurrences) was above a predefined threshold. The optimal threshold level was found to be 0.6 on the training set using F-measure as the objective function. The resulting terms were then matched in unseen text, providing the basis for the span expansion.

To expand head term spans mandated by AG, we previewed several parsers' output to find out which of them best conformed to the examples enlisted in the AG. We compared the Stanford Parser[2], Charniak-Lease parser with the biomedical language model from McClosky[3], the Enju parser,[4] and GENIA Tagger's chunker[5]. On grammatically correct sentences, we found Stanford Parser to most closely conform to the examples in AG, while GENIA chunker performed best on grammatically loose sentences. A common disagreement between parser outputs and the AG examples was that parsers tended not to split noun phrases on connectives (CC) thus in a sentence like:

"The patient experienced X and Y.",

where the enclosing noun phrase of the term *X* was "*X"* according to the AG examples, while *"X and Y"* according to the most parsers. To circumvent this behavior, we split noun and adjective phrases on connectives. Another disagreement was the inclusion of certainty expressing phrases (e.g., *presumed*, *likely*) into NPs by the parsers, which though linguistically correct was rejected by AG for obvious reasons (see Section Assertion classification).

We implemented the AG requirements as graph matching rules on the parse trees. The rules expanded the matched head terms to the containing noun or adjective phrases, and further extended the phrases by the appropriate PPs (prepositions). We combined the expansions obtained from Stanford Parser and the GENIA chunker outputs by taking their union.

Though the matched head terms are non-overlapping, their spans obtained through the expansion may overlap. E.g., if the dictionary for the "problem" class contained both "reflux" and "disease", but not "reflux disease" the sentence "The patient has reflux disease." would have two head terms ("reflux" and "disease") and the expanded span "reflux disease" twice (once for each head term). To prevent such possibly confusing output, we merged all overlapping spans of the same class.

Assertion classification

The assertion classification task was solved by a rule based system, which analyzed the context of the problem mentions to classify their status. As defined by the AG, the status (assertion class) can be either *Present*, *Absent*, *Possible*, *Hypothetical*, *Conditional* or *Associated with someone else*.

A problem mention's context was broken down into partially overlapping scopes: the containing *sentence*, words *before* the mention, words *after* the mention, words *inside* the mention, words in a fixed width window *around* the mention. We identified trigger terms indicating an assertion class and specified for each one in which scope it should occur. For examples trigger terms, see Table 1.

Status	Before	After	Around
Present	due to		
Absent	without		Non
Possible	possible		
Hypothetical	if you		Any
Conditional		when	Exertion
Someone else	wife		

Table 1. Example trigger terms for assertion classes.

[2] http://nlp.stanford.edu/software/lex-parser.shtml
[3] ftp://ftp.cs.brown.edu/pub/nlparser/reranking-parserAug06.tar.gz
[4] http://www-tsujii.is.s.u-tokyo.ac.jp/enju/
[5] http://www-tsujii.is.s.u-tokyo.ac.jp/~genia/postagger/geniatagger-3.0.1.tar.gz

Type	Statistics				Methods					
	Positive	**Negative**	**Pos/Neg**	**Pos %**	**kBSPS**	**SpT**	**SL**	**PT**	**APG**	**n-gram**
TERP	1711	1858	0.92	48%	0.78	0.74	**0.83**	0.74	**0.83**	0.68
TRAP	1413	2874	0.49	33%	0.64	0.64	0.71	0.64	**0.74**	0.52
PIP_FW	900	7688	0.12	10%	0.49	0.00	**0.55**	0.00	N/A	0.29
PIP_BW	320	8268	0.04	4%	0.17	0.00	**0.27**	0.00	N/A	0.12
TECP	295	3274	0.09	8%	0.49	0.44	**0.51**	0.41	0.45	0.35
TRCP	294	3993	0.07	7%	**0.45**	0.34	0.42	0.35	0.43	0.29
TRIP	107	4180	0.03	2%	**0.40**	0.23	0.38	0.24	0.28	0.37
TRNAP	106	4181	0.03	2%	**0.44**	0.27	0.37	0.23	0.37	0.27
TRWP	56	4231	0.01	1%	**0.19**	0.13	0.02	0.13	0.11	0.11
Overall	**5202**	**40547**	**0.13**	**11%**	**0.60**	**0.47**	**0.65**	**0.47**	**0.52**	**0.54**

Table 2. Relation type statistics and relation extraction cross-validation results on the training set. Positive examples are relations in the training data, negative examples are two concepts of the correct type in the same sentence without a corresponding relation. Micro F-measure values, best result for each type in bold.

We also re-used our context semantics identification system [7] designed for the past i2b2 Obesity Challenge, which provides both phrase and section level negation, allergy and family history annotations. Problem mentions contained in one of these annotations were marked as being *Absent*, *Conditional* and *Associated with someone else* respectively.

Disagreements between assertion classes assigned by the different methods were resolved by favoring the assertion class with higher *prior* probability measured on the training set. A problem mention, for which no assertion class could be determined by the above process, was classified as *Present*.

Relation extraction

The relation extraction task of the challenge can be considered as a multiclass classification task. We solved this problem using a set of binary classifiers (one for each of the 7 symmetric relation types and 2 for the asymmetric PIP). Finally, we resolved the ambiguous cases, that is, when several classifiers classified a pair as a positive w.r.t the corresponding relation type. Note that concept types limit the possible relations: there are 5 relationship types between medical problems and treatments, 2 between tests and medical problems, and the asymmetric PIP – that is considered as two types, PIP_FW and PIP_BW (forward and backward) – between two medical problems. Observe that all relation types include at least one medical problem concept. Statistics of the relation types are given in Table 2.

We applied two different approaches to tackle the relation extraction task. As a baseline, we used a co-occurrence based method, which identifies frequent words and word n-grams between concepts in the training set for each relation type, and classifies test examples based on the thus discovered patterns.

The second approach cast the problem into a machine learning scenario, and applies support vector machines (SVMs) with different kernel functions to learn a classification model train for each relationship type.

In order to evaluate machine learners with cross-validation (CV), we defined a 10-fold split on the training data. For comparison, we also used these folds to evaluate the word n-gram based approach.

Word n-gram based approach

The (word) n-gram based classifier works based on the occurrence statistics of continuous word sequences in the training data. We built a model for each relation type.

We tokenized the sentences into 4 types of tokens: entities (defined in training data), numbers (words containing only digits), other words and punctuations. We blinded entities into their types and ignored numbers' actual values. We created an n-gram dictionary by indexing the training set. Here we experienced with several parameters relating the length of the n-gram (min and max) and its overall minimum occurrence (OMO).

We quantify the usefulness of an n-gram relating to (typed) relation extraction by assigning a precision value (for each type): the ratio of positive to all

Task		Description	TP	FN	FP	R	P	F
Concept	sub2	Dictionaries + span expansion†	24 892	20 117	16 962	0.55	0.59	0.57
Assertion	sub4	Trigger terms + context semantics*	15 805	2 745	0	0.85	1.00	0.92
Relation	sub1	SL (shallow linguistic kernel)*	6 301	2 769	3 122	0.69	0.67	0.68
	sub2	kBSPS (k-band shortest path kernel)	447	8 623	3 772	0.05	0.11	0.07
	sub3	Word n-gram based*	6 040	3 030	23 697	0.67	0.20	0.31
	sub4	SL+kBSPS*	4 639	4 431	8 636	0.51	0.35	0.42

Table 3. Evaluation results on the test set, challenge submissions indicated by *, concept evaluation method is "exact span, exact class" assertion evaluation method is "matching span". †not submitted due to administrative problems.

sentences where the n-gram occurs. An n-gram containing entity in a positive sentence was counted as positive only if the entity was in the tagged relation.

At classification a sentence was deemed as positive if the highest precision of contained n-grams' reached a predefined threshold. Obviously, here we only considered n-grams in the dictionary. To calculate the precision we applied the abovementioned 10-fold CV and used 10% of each fold to set the threshold.

For the parameters of n-grams, we found that n=1,…,4 and OMO=4 is the best choice according to the average F-measure calculated on the 10-fold CV. We also experienced with other aggregating functions in addition to maximum (at sentence level classification) but all appeared to be inferior.

Kernel based approaches

A support vector machine (SVM) is a classifier that, given a set of training examples, finds the linear (hyper)plane that separates positive and negative examples with the largest possible margin [8]. If the two sets are not linearly separable, kernel functions can transform the problem space to a nonlinear, often higher dimensional space, in which the problem might be separable [9]. The kernel is a similarity function that maps a pair of instances to their similarity score. Kernels can be easily computed with inner products between instances, which allows the use of high dimensional feature spaces such as the rich structured representation of graphs or trees.

We made use of our kernel comparison framework [10], devoted originally to PPI relation extraction. From the 13 available kernels in the package, we experimented here with 5: shallow linguistic (SL) [11], partial tree (PT) [12], spectrum tree (SpT) kernel [13], *k*-band shortest path spectrum (kBSPS) [10, 14], and all-paths graph (APG) [15] kernels. Before doing that we had to preprocess the provided discharge summaries to be compliant with the kernels.

The selected kernels use different sentence representations, namely POS-tag sequence, syntax tree or dependency graph. For PT kernel, we parsed the sentences with Charniak-Lease parser[6] using the language model from McClosky trained on biomedical texts [16]. For SL, we additionally applied GENIA Tagger[7] to obtain lemmas. The resulted syntax trees were transformed to dependency graphs – needed for kBSPS and APG kernels – using the Stanford converter.[8] All parse results were converted to the XML-format of [17], which is the *de facto* standard for PPI relation extraction.

We experimented with different parameter settings for each kernel at the CV evaluation to identify the best one, which is then used to train a model on the entire corpus (no CV) and apply it on unlabelled test documents.

We compare in Table 2 the results achieved by the kernel based SVMs and the baseline n-gram method using the 10-fold CV setting. For each kernel we indicate only the results with best settings.

Results

Table 3 summarizes the evaluation results on the test set as measured by the evaluation program provided by the organizers. Note that official evaluation results were not available prior to the submission of this paper. Table 4 presents detailed evaluation of the concept identification task. Per class assertion classification results can be found in Table 5.

[6] ftp://ftp.cs.brown.edu/pub/nlparser/reranking-parserAug06.tar.gz
[7] http://www-tsujii.is.s.u-tokyo.ac.jp/~genia/postagger/geniatagger-3.0.1.tar.gz
[8] http://nlp.stanford.edu/software/lex-parser.shtml

Exact	Class	TP	FN	FP	R	P	F
Exact	Problem	11.4k	7.2k	5.4k	0.61	0.68	0.64
	Treatment	7.3k	6.3k	5.6k	0.54	0.57	0.55
	Test	6.3k	6.6k	6.1k	0.49	0.51	0.50
	Overall	**24.9k**	**20.1k**	**17.0k**	**0.55**	**0.59**	**0.57**
Inexact	Problem	14.2k	4.3k	2.1k	0.77	0.87	0.82
	Treatment	10.3k	3.3k	2.4k	0.76	0.81	0.79
	Test	10.0k	2.9k	1.8k	0.77	0.84	0.81
	Overall	**36.3k**	**8.7k**	**6.3k**	**0.77**	**0.85**	**0.80**

Table 4. Concept identification results on the test set using the "exact class" evaluation method.

Class	TP	FN	FP	R	P	F
Present	11 754	1 271	0	0.90	1.00	0.95
Absent	2 934	675	0	0.81	1.00	0.90
Possible	596	287	0	0.67	1.00	0.81
Hypothetical	361	356	0	0.50	1.00	0.67
Conditional	26	145	0	0.15	1.00	0.26
Someone else	134	11	0	0.92	1.00	0.96
Overall	**15 805**	**2 745**	**0**	**0.85**	**1.00**	**0.92**

Table 5. Assertion classification results on the test set using the "matching assertion" evaluation method.

Discussion

In the concept identification task, the 0.23 gap between exact and inexact F-measures seen in Table 4 signifies the difficulty of correctly assigning the spans to trigger terms. Precision being constantly higher than recall on the test set is a result of dictionary optimization for F-measure on the training set.

As for the assertion classification task, the classification performance was correlated with the number of available training examples, see Table 5. The only exception is the class Associated with someone else, which was easier to recognize due to the presence of unambiguous indicator phrases, such as "family history". The official evaluation program did not report any false positives, which is rather misleading, see the overview paper by the organizers for a hopefully more accurate evaluation.

At relation extraction, after evaluating the methods in Table 2, we found that – in accordance with the findings of [10] – SpT and PT kernels are inferior to the other three (kBSPS, SL and APG), therefore we disregarded them in further evaluation. The available implementation of APG is slower by a factor of 10-50 (depending on the corpus) which made difficult the experimentation within short time. Therefore for submissions, we selected the classification models of kBSPS and SL kernels using their best settings. Surprisingly, kBSPS produce only very few positive prediction, which resulted in a much worse F-score. The gap between the F-score of SL and kBSPS is extremely large. One reason could be that SL is more robust (this is the only kernel basically without any parameters, which is clearly an advantage, see [18]), while kBSPS exhibits many parameters. Alternatively, it might be also caused by a yet undiscovered implementation error.

Conclusion

In this paper we presented our approaches for the i2b2 Relation Challenge. For the concept classification tasks, we used dictionaries to locate the clue words and grammatical information (parse tree based) to determine its scope. At assertion classification, we applied a rule-based scope identifier for assertion clue words. For the relation extraction, we used a baseline n-gram base method and SVMs with kernel functions. In each task we could achieve state-of-the-art results, according to the provided evaluation tool.

References

1. Krallinger M, Erhardt RA, Valencia A. Text-mining approaches in molecular biology and biomedicine. Drug Discov Today. 2005; 10(6):439-45.
2. Farkas R, Szarvas G. Automatic construction of rule-based ICD-9-CM coding systems. BMC Bioinformatics. 2008; 9(S3):S10.
3. Plake C, Schiemann T, Pankalla M, Hakenberg J, Leser U. ALIBABA: PubMed as a graph. Bioinformatics, 2006; 22(19):2444-5.
4. Ideker T, Sharan R. Protein networks in disease. Genome Res. 2008; 18: 644–652.
5. Cohen AM, Hersh WR. A survey of current work in biomedical text mining. Brief Bioinform, 2005; 6(1):57-71.
6. Kabiljo R, Clegg A, Shepherd A. A realistic assessment of methods for extracting gene/protein interactions from free text. BMC Bioinformatics. 2009; 10: 233.
7. Solt I, Tikk D, Gál V, Kardkovács ZT. Semantic classification of diseases in discharge summaries using a context-aware rule-based classifier. J Am Med Inform Assoc. 2009; 16:580–4.

8. Joachims T. Making large-scale support vector machine learning practical, Advances in kernel methods: support vector learning. Cambridge, MA: MIT Press; 1999.
9. Schölkopf B, Burges CJC, Smola AJ, editors, Advances in kernel methods: support vector learning. MIT Press; 1999.
10. Tikk D, Thomas P, Palaga P, Hakenberg J, Leser U. A comprehensive benchmark of kernel methods to extract protein-protein interactions from literature. PLoS Comp Biol, 2010; 6(7): e1000837.
11. Giuliano C, Lavelli A, Romano L. Exploiting shallow linguistic information for relation extraction from biomedical literature. In: Proc. of the 11th EACL'06, Trento, Italy: The Association for Computer Linguistics, 2006; pp. 401–408.
12. Moschitti A. Efficient convolution kernels for dependency and constituent syntactic trees. In: Proc. of the 17th European Conf. on Machine Learning, 2006; Berlin, Germany, pp. 318–329.
13. Kuboyama T, Hirata K, Kashima H, Aoki-Kinoshita KF, Yasuda H. A spectrum tree kernel. Information and Media Technologies 2007; 2: 292-299.
14. Palaga P. Extracting relations from biomedical texts using syntactic information. Master's Thesis, Technische Universität Berlin, 2009.
15. Airola A, Pyysalo S, Björne J, Pahikkala T, Ginter F, Salakoski T. All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. 2008; BMC Bioinformatics 9: S2
16. McClosky D. Any domain parsing: automatic domain adaptation for natural language parsing. Ph.D. thesis, Department of Computer Science, Brown University, 2009.
17. Pyysalo S, Airola A, Heimonen J, Björne J, Ginter F, Salakoski T. Comparative analysis of five protein-protein interaction corpora. BMC Bioinformatics 2008; 9 Suppl 3: S6.
18. Keogh E, Lonardi S, Ratanamahatana CA. Towards parameter-free data mining. In: Proc. of the 10th ACM Int. Conf. on Knowledge Discovery and Data Mining (KDD), Seattle, WA, USA: ACM, 2004; pp. 206-215.