Biomedical Informatics Insights

ORIGINAL RESEARCH

Combining an Expert-Based Medical Entity Recognizer to a Machine-Learning System: Methods and a Case Study

Pierre Zweigenbaum[1], Thomas Lavergne[1,2], Natalia Grabar[3], Thierry Hamon[4], Sophie Rosset[1] and Cyril Grouin[1,5]

[1]LIMSI-CNRS, Orsay, France. [2]Université Paris-Sud 11, Orsay, France. [3]STL CNRS UMR 8163, Université Lille 1 et 3, Villeneuve-d'Ascq, France. [4]LIM&Bio (EA3969), Université Paris 13, Bobigny, France. [5]INSERM U872 Eq 20 and UPMC, Paris, France. Corresponding author email: pz@limsi.fr

Abstract: Medical entity recognition is currently generally performed by data-driven methods based on supervised machine learning. Expert-based systems, where linguistic and domain expertise are directly provided to the system are often combined with data-driven systems. We present here a case study where an existing expert-based medical entity recognition system, Ogmios, is combined with a data-driven system, Caramba, based on a linear-chain Conditional Random Field (CRF) classifier. Our case study specifically highlights the risk of overfitting incurred by an expert-based system. We observe that it prevents the combination of the 2 systems from obtaining improvements in precision, recall, or F-measure, and analyze the underlying mechanisms through a post-hoc feature-level analysis. Wrapping the expert-based system alone as attributes input to a CRF classifier does boost its F-measure from 0.603 to 0.710, bringing it on par with the data-driven system. The generalization of this method remains to be further investigated.

Keywords: natural language processing, information extraction, medical records, machine learning, hybrid methods, overfitting

Biomedical Informatics Insights 2013:6 (Suppl. 1) 51–62

doi: 10.4137/BII.S11770

This article is available from http://www.la-press.com.

Introduction

In the medical domain, a wealth of documents is produced for each patient, such as a letter from the attending physician, hospital entrance document, laboratory analysis results report, hospitalization records, nurse records, and so on. All those documents convey clinical information that can be useful for a health care professional to treat the patient or to perform clinical research. However, their textual nature locks clinical information into these documents, preventing it from being included in further processing workflows. Systems that extract information from clinical texts are thus needed.[1] Automatically detecting clinical concepts and, more generally, medical events related to a patient is the first level of clinical information extraction from medical texts.

Most work on information extraction falls into 2 broad kinds of methods. Expert-based methods rely on human knowledge, typically including lexicons, patterns and rules, to detect entities.[2,3] Data-driven methods rely on data, generally in the form of annotated corpora, to induce knowledge or decision procedures to perform entity detection. As each of these methods has its advantages and limitations, combining them together may overcome their individual limitations and lead to improved results. Many methods can be used for this purpose, and various authors have focused on some of them. We present here a case study where an existing expert-based medical entity recognition system, Ogmios,[4] is combined with a data-driven system, Caramba,[5] based upon a linear-chain Conditional Random Field (CRF) classifier. We examine different methods to combine 2 such systems and test the most relevant methods through experiments performed on the i2b2/VA 2012 challenge data.

In this paper we first outline the overall process of entity recognition in texts, with an eye on clinical information extraction. We examine issues encountered when combining expert-based and data-driven methods and determine the most relevant combinations for our setting. We then describe in more detail the datasets on which we performed experiments and the 2 systems we start from. We finally present the results obtained in these experiments and propose an analysis of how the expert-based system influences the final decisions. We summarize our findings in the Conclusion section.

Entity Recognition: From Information to Decision

Named Entity Recognition, as defined in the MUC conferences,[6] aims to detect names of persons, locations and organizations in texts. More generally, we use the term Entity Recognition to cover the detection of entities in a domain, not only expressed with proper names, but also with other parts of speech. In the clinical domain, this includes among others signs, symptoms and diagnoses (medical PROBLEM), medications and surgical interventions (TREATMENT) and hospitals and clinical departments (CLINICAL_DEPT).

Detecting entities, be it through expert-based or data-driven methods, relies on the collection of information from the input text. This information spans basic typographical properties such as upper or lower case, as well as numeric or alphabetic characters, to document structure such as its division into sections (eg, history of present illness, hospital course). External knowledge and components are often applied to the text to normalize it (eg, lemmatization), to determine in context word properties such as parts-of-speech or semantic classes, or to add structure to it (eg, syntactic dependencies), based on resources such as lexicons and thesauri, and/or Natural Language Processing components such as part-of-speech taggers and syntactic parsers.

Based on this information, decisions must be made as to whether or not a given type of entity occurs at a given position in the text. A common decision paradigm consists of eliciting rules which, given the information attached to selected parts of a sentence, infer the presence of an entity. For instance, a pattern such as '*the* <*Maj*> *hospital*' (where <*Maj*> stands for a capitalized or uppercase word) can be used to build a rule which infers the presence of a CLINICAL_DEPT entity. Such rules are usually categorical (Boolean), ie, they make binary decisions. Another decision paradigm learns a decision procedure by observing texts (generally associated with annotations): this is the (supervised) learning paradigm. It accounts for the majority of works currently, where it is usually based on statistical machine learning. We focus here on a model that belongs to the discriminant family of machine-learning models: linear-chain Conditional Random Fields (CRF),[7,8] a log linear model which is particularly relevant for sequences, hence

for sentences. Schematically, the log-linear model learns a set of 'soft' (weighted) rules, in the form of weighted feature functions, which vote according to their weight for or against each of the target classes. These classes are the target entity types (eg, PROBLEM) or more usually a variant where Begin, Inside, and Outside positions (B-I-O) are distinguished in a target entity of that type: eg, B-PROBLEM (first token in a PROBLEM entity), I-PROBLEM (any other token in a PROBLEM entity), and O (outside any entity). Given the non-null feature functions for a token, the sum of positive and negative weights of these features plays a central role in the computation of the prediction score of a given entity type for this token.

Combining Expert-Based and Data-Driven Methods

As mentioned in the Introduction, many works combine expert-based and machine-learning methods with the aim of overcoming their individual limitations and improving their results. For instance, in the 2010 i2b2/VA challenge on 'concept extraction',[9] 3 of the 10 top-performing systems were categorized as 'hybrid', 5 were labelled 'supervised' and 2 'semi-supervised.' When looking more closely, most of the 'supervised' or 'semi-supervised' systems used information obtained by applying expert-based systems such as MetaMap[10] or cTAKES,[11] hence should also receive the 'hybrid' label.

There exist many ways to combine 2 systems implementing different methods. In this paper, we are specifically interested in combining an expert-based and a machine-learning system. We consider in turn the following combination schemes: union and intersection, primary system plus fallback, voting, and using the expert-based output as a feature in a classifier.

Union and intersection

Union and intersection are the simplest combination schemes. They assume there is a 'null' class, the O class in the B-I-O encoding, which means 'no entity', and is handled differently from the other (B-I) classes.

Intersection of the outputs of the 2 systems means that a decision is made only when both systems produce the same non-null output. In the other cases, the null class (O) is produced. Intersection can result in illegal O-I sequences which must be repaired (any entity must begin with a B class). Union of the outputs means that the non-null outputs of both systems are kept. This is applicable only if both systems produce compatible outputs, ie, they never produce 2 different non-null classes for the same token. Otherwise some other combination scheme must be used.

These combinations are rather crude and in our case they resulted in poorer results than the initial systems.

Primary system plus fallback

This combination also makes a distinction between the null O class and the other classes. It considers 1 of the 2 systems as primary and systematically trusts its B-I classes. The other system is used as a fallback and is consulted only when the primary system outputs an O class. This is particularly relevant when the primary system has higher precision and but low recall: the fallback system can increase recall while leaving the (mostly) correct decisions of the first system intact. The expert-based system is often the one with the higher precision and thus plays the role of the primary system.

Here again, this combination may result in inconsistencies: eg, if the primary system outputs a B-TEST I-TEST O sequence and the fallback system outputs an I-PROBLEM for the O token, this results in an illegal B-TEST I-TEST I-PROBLEM sequence.

'Forced decoding' is a better way of implementing this combination scheme when the fallback system is a Conditional Random Field (CRF) (this is however applicable to some other classifiers). It provides the output of the primary system to the CRF and imposes its choice of non-null classes both during the training and inference stages. Since these imposed choices are known beforehand in a given sentence, choices for the other tokens of the same sentence can take them into account. This avoids the above-mentioned inconsistencies and results in more relevant choices overall.

Previous work has reported good results with forced decoding. For instance, for Arabic named entity recognition, Gahbiche-Braham et al[12] combined a very precise expert-based system to a CRF based on other features. When imposing through

forced decoding to the CRF the entities found by the first system, the F-measure of the CRF increased to 0.84, significantly outperforming both the expert-based system (F = 0.74) and the CRF with its own features (F = 0.73).

In our case study, forced decoding was not suitable because the precision of the expert-based system, while reasonably good (0.82), was not high enough.

Voting

Voting is useful when several systems are to be combined. Majority voting takes the decision proposed by the highest number of systems. With 2 systems as in our case study though, majority voting amounts to intersection as presented above. It is more relevant with many systems.

Majority voting gives each system a weight of 1. In weighted voting, weights can be different from 1; these weights may be based on a measure of confidence attached to each system's decisions. Many data-driven methods can provide confidence estimates; this is generally not the case however of expert-based systems.

A workaround consists in wrapping the expert-based system within a classifier and learning confidence estimates from a training corpus. However, in our case, the training corpus was already used to develop and tune the expert-based system. For obvious reasons we could not do that on the test corpus either. Therefore we had no means to compute confidence estimates reliably for the expert-based system.

Using the expert-based output as a feature in a supervised classifier

A generalization of the last approach consists of using the outputs of the systems as features in a supervised classifier. This is more relevant than voting if there are only few systems. The second-stage classifier (eg, a log-linear model) is then trained and typically learns weights for each input classifier. In our case study, the data-driven system relies on a log-linear model (a CRF). Therefore there is no real need to separate 2 stages: we can just add a feature for the expert-based system to the data-driven system.

This general method is particularly relevant when using a discriminant model, because such models can use features that are not independent—and it is to be hoped that the systems will agree in many cases. In contrast, a generative model would need to take into account dependences when modelling observations.

This is the most convenient method, and we shall illustrate it below. Note however that the same issue as above is raised about the training corpus. We shall therefore study how it shows in our results.

The i2b2/VA 2012 Challenge and Corpus

Our case study uses the datasets of the i2b2/VA 2012 challenge[13] (https://www.i2b2.org/NLP/, i2b2—Integrating Informatics for Biology and the Bedside, Albany, NY). It is restricted to the detection of 'events' in clinical texts (the challenge also involved other tasks, including the detection of temporal expressions, polarity and modality, and temporal links). The i2b2 evaluation campaign is an international text mining challenge that focused on the clinical domain. From 2010 to 2012, the participants were asked to extract several clinical concepts of different kinds. Problem, test and treatment were common to these 3 editions. Additional annotations were progressively added over the years, including identification of 'assertions' (polarity and modality) and relations between concepts in 2010,[9] co-reference between concepts in 2011,[14] and finally events, polarity and modality, time expressions, and temporal relations between concepts and time expressions in 2012.

The clinical documents of the 2012 corpus come from the following organizations: Beth Israel Deaconess Medical Center (Boston, MA); Partners Healthcare (Boston, MA); and University of Pittsburgh Medical Center (Pittsburgh, PA) (they were also present in the 2011 corpus).

Definition of the task: event detection

The present work focuses on the extraction of clinical events as defined in the i2b2 NLP challenge held in August 2012, to which we participated. These events are different both from existing research and from the previous editions of the i2b2 challenges. 6 types of events were defined as follows:

1. Medical problems, which include the patient's complaints, symptoms, diseases and diagnoses:
   - The patient reportedly had *chest pain*. She reported some *shortness of breath*. His arterial blood gas showed a *respiratory acidosis* with a PCO2 of 71.

2. Tests, which include any clinical lab tests, exams, and their results:
   - His *arterial blood gas* showed a respiratory acidosis with a PCO2 of 71. A *CT scan* showed that the patient has ...
3. Treatments, which include medications, surgeries and other procedures:
   - The patient was taken to the operating room and underwent an *orthotopic liver transplant*. On 8/30/2001, the patient was *extubated* in the operating room. *Ativan* p.r.n. was given for this.
4. Clinical departments, which include the clinical units involved in the patient's treatment.
   - The patient was taken to the *operating room* and underwent an orthotopic liver transplant. The patient underwent an uncomplicated recovery in the *intensive care unit*.
5. *Evidential* markers specify the source of information:
   - The patient *reportedly* had chest pain. His arterial blood gas *showed* a respiratory acidosis with a PCO2 of 71. He *complains* of headache.
6. *Occurrence* is the default value for event types. It is used for all the other kinds of clinically relevant events that occur/happen to the patient:
   - He was *readmitted* for sternal wound infection.

Annotated corpora

The corpus provided by the i2b2 challenge contains 210 clinical records. These records are divided into training (190) and test (120) subsets. The distribution of the events is similar across the training and test corpora (Table 1). There is, however, a largely unbalanced distribution of event types: treatments (TTT) and medical problems (PRB) represent more than half of the annotations, while clinical departments (Dept) and evidential markers (Evid) have very few occurrences in the corpora.

The reference annotations were provided by the i2b2 organizers: each clinical record was annotated manually by 2 experts. 8 experts participated in this task. The annotated training set was available before the challenge dates, and the bare test set was provided at the date of the challenge. The reference annotations for the test set were disclosed after the challenge was closed.

Two Event Detection Systems

We started from 2 existing systems which be present below:

1. An expert-based medical entity recognition system, Ogmios;[4]
2. A data-driven system, Caramba,[5] based upon a Conditional Random Field (CRF) classifier.

Expert-based method: Ogmios

Overall architecture

The Ogmios platform proposes several standard NLP functionalities and can be easily modified to new tasks, domains and applications. In the current experiments, its configuration is similar to those defined for the former i2b2 2009 and i2b2/VA 2010 Challenges:[4,15]

1. POS tagging is performed with GeniaTagger;[16]
2. Event identification relies on the TermTagger Perl module (http://search.cpan.org/~thhamon/Alvis-TermTagger/) and the linguistic and semantic resources described below;
3. A term extraction system, YaTeA,[17] is applied to detect the noun phrases;
4. A specific post-processing adapted to the i2b2/VA 2012 event definition selects and extends previously identified events.

The post-processing step first extends event strings to the right with acceptable POS tags, stopping with the first non-acceptable stopword. Then, it selects the final event according to several criteria: (i) in case of competing annotations, the larger event string is preferred; (ii) an event identified as a non-i2b2 concept (as well, M.D., etc.) is rejected; (iii) events occurring in section titles are removed except for the tests (Serologies) and occurrences (Discharge Date); (iv) contextual tags (see below, Linguistic resources)

Table 1. Annotation statistics in percentage on training and test corpora for the six clinical event types.

	Clinical_dept	Evidential	Occurrence	Problem	Test	Treatment
Train	6.05	4.49	19.95	30.50	15.76	23.25
Test	5.39	4.38	18.38	31.70	15.99	24.17

are used to semantically label the extracted events; (v) section titles are also used to categorize events (for instance, the Admission Diagnosis section usually contains medical problems); (vi) the extracted noun phrases are used to detect or adjust syntactically the boundaries of the clinical events.

Linguistic and semantic resources

We exploited the following main terminological resources:

- 316,368 terms from the UMLS[18] which belong to several semantic axes related to the involved types of events:
  1. medical problems (B2.2.1.2.1 Disease or Syndrome, A2.2.2 Sign or Symptom, B2.3 Injury and Poisoning, A1.2.2 Abnormality and A1.1.5 Bacteries),
  2. tests (B1.3.1.1 Diagnostic procedures and B1.3.1.2 Laboratory procedures),
  3. treatments (B1.3.1.3 Therapeutic or prevention procedures,
  4. clinical departments (A2.7.1 Health Care Related Organization).
- 243,869 entries from RxNorm[19] used for the detection of medication names (treatments);
- The available annotations of the 2012 i2b2 training sets;
- Additional contextual tags for marking specific contextual clues for different types of events (pre-problem, pre-test, pre-treatment, post-treatment ...).

These resources were manually checked and adapted to increase their coverage and precision. The contextual tags were built specifically for this purpose.

Data-driven method: Caramba

The Caramba system[5] relies on several tools that were used to compute various input features, shown in Table 2. These features were input to a Conditional Random Field (CRF) classifier, Wapiti,[20] (http://wapiti.limsi.fr/) through 'patterns,' as in most CRF classifiers. Patterns specify which attributes or combinations or attributes of the current token and of other tokens in the current sentence are used, together with the class of the current token (or with a bigram of this class and the class of the previous token), to build feature functions.

In our experiments on the training corpus, we tested models generating up to 155 million features. Since CRFs may be prone to overfitting (eg, compared to SVMs), we took care to include features that would lead to better generalization: syntactic tags and chunks, and semantic classes of various kinds.

Evaluation of the individual systems

We measured the precision, recall and F-measure of these event detection systems in 2 conditions:

1. Directly, through a strict comparison to the gold standard annotations, as computed by the conlleval.pl program (http://www.clips.ua.ac.be/conll2000/chunking/).

Table 2. Features for CRF-based event identification.

- Section id among four sections we defined as follow: admission date (section #1), discharge date (#2), history of present illness (#3) and hospital course (#4);
- Morpho-syntactic tagging with the Tree Tagger (http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/)[21] and home-made noun phrase chunking based upon the previous tags;
- Morpho-syntactic tags projected from a specific lexicon of 62,263 adjectives and 320,013 nouns based on the UMLS Specialist Lexicon;
- Semantic types and semantic groups from the UMLS (http://www.nlm.nih.gov/research/umls/);
- Semantic annotation (the six event types and other markers such as "anatomical part", "localization", "pre/post-examination", "value unit", etc.) with WMatch,[22] an analysis engine based upon regular expressions of words, rules and lexicons;
- Syntactic analysis with the Charniak McClosky biomedical parser (http://stanford.edu/~mcclosky/biomedical.html):[23] we used part-of-speech and chunk information derived from the parse trees;
- Two series of unsupervised clusters obtained through Brown's algorithm,[24] performed over the UMLS Metathesaurus[18] terms (multi-words expressions) and over the 2011 i2b2/VA Beth Israel and Partners Healthcare corpora. This corpus was selected because it was closest to the 2012 training corpus. It is probable that some test documents belong to this corpus, however since this processing is unsupervised and performed on the unannotated documents, it is good practice to do it (we should even have performed it again once we had the full set of unannotated test documents). Clustering was performed with code from Liang's Master's Thesis[25] (http://www.cs.berkeley.edu/~pliang/software/).

2. With the evaluation program of the i2b2/VA challenge (i2b2Evaluation.py), in its default 'overlap' mode, which counts as correct overlapping entities with possibly different types.

Experiments were performed on the training corpus. When it concerned the data-driven system, this was done with 10-fold cross-validation. A model was then learnt on the whole training corpus, and then applied to the test corpus.

Ogmios was evaluated as is (see Table 3).

When training Caramba on the training corpus, we examined which groups of features were most discriminant. For this purpose, elementary subsets of features were defined by groups of patterns according to Table 2. We trained and tested the CRF with increasing subsets of the whole set of features, following a greedy approach, with 10-fold cross-validation. For each test fold, 8 of the remaining folds were used for training and the remaining 1 was used as a development set that the classifier used to compute its error rate. The number of training iterations was limited to 30 to speed up the comparison of feature subsets. Starting from scratch, each group of patterns was tried independently.

At the first iteration, 8 groups of patterns obtained an F-measure (averaged over the 10 folds) greater than 0.0001 (see Table 4). It is noticeable that unsupervised features (Brown clusters) based upon the i2b2 clinical corpus or upon the UMLS terms obtain quite a high F-measure (resp. 0.6859 and 0.6082) without any other help. We added all these groups of patterns to our pool of patterns. This increased the F-measure to 0.7124 by only 2.7 points compared to the i2b2 Brown clusters.

In a second iteration, we independently tested the addition of each remaining group of patterns to this pool (Table 5). Adding all these pattern groups to the pool resulted in 101,492,274 features and an improved F-measure of 0.7317. We also tested the addition of only a subset of these pattern groups, selecting the less redundant ones (see last line of Table 5). This slightly improved the F-measure again (0.7323) and reduced the number of features to 15 million. Note that these settings involve further work beyond the system that we presented at the challenge, hence the results reported here are better than those we obtained in the challenge.[13]

Table 3. Ogmios: direct evaluation (training set).

	P	R	F	Description
Ogmios	0.8229	0.7079	0.7611	Ogmios as is

Abbreviations: P, Precision; R, Recall; F, F-measure.

Combination of the 2 Methods

Tested combinations

We tested the combination of the expert-based system Ogmios with the data-driven method Caramba by including the output of Ogmios as attributes provided to the CRF classifier used in Caramba. Development was performed on the training set and monitored with the conlleval.pl evaluation program (left set of P-R-F columns in Table 6: Training, conlleval).

Ogmios was first used as the only attribute in the CRF, providing 2 information items: the target event type (6 possible values + no event), and specific semantic markers from the previous challenges such as "dosage", "duration", "mode of administration" (i2b2 2009), contextual tags ("pre/post-possible", "pre/post-conditional", "pre/post-problem", "pre/post-treatment", "pre/post-negation", "pre/post-proposed", etc.), polarity/modality and concept markers from former and this year's challenges. Each of these 2 information items was encoded with the B-I-O scheme. This corresponds to wrapping Ogmios within a classifier, which was trained on the training corpus. Table 6 (row OgF) shows the obtained results (for ease of comparison, Table 6 recalls the evaluation of Ogmios (row Og) and Caramba (row Ca) alone). We can see that wrapping Ogmios in the classifier as unigrams and bigrams of attributes, trained on the training corpus, substantially increases precision (+5 pt)) but decreases recall (−1 pt), resulting in an increase in F-measure (+2 pt).

Ogmios attributes were then complemented with those selected for Caramba and presented in Table 5. Table 6 (row OgCa) shows that both precision (+15 pt) and recall (+2 pt) are much improved over when Ogmios is used as only attribute (row OgF). This is our best result on the training set, and improves precision, recall and F-measure both over Ogmios and over Caramba alone. Besides, precision and recall are better balanced.

We also examined the combination of the Ogmios attributes with various subsets of Caramba's

Table 4. Caramba: best groups of patterns at first iteration (training set).

P	R	F	Description
0.7322	0.6452	0.6859	B: Brown Beth_Partners unigrams
0.6949	0.5407	0.6082	Brown UMLS unigrams
0.5239	0.3475	0.4179	B: UMLS first or two Semantic Types
0.4307	0.2908	0.3472	Charniak-McClosky POS unigrams, bigrams, trigrams
0.5209	0.2551	0.3425	Wmatch only
0.5137	0.2564	0.3421	Wmatch only, BIO
0.3378	0.1476	0.2055	B: Charniak-McClosky chunk unigrams, bigrams, trigrams
0.3378	0.0478	0.0837	B: alphabetic or case unigrams
0.7469	0.6809	0.7124	Subset 1: All of the above

Note: B: bigram of classes.

attributes and discovered that the addition of very basic attributes, namely the normalized (lowercased) token was enough to boost the results of Ogmios (row OgT): it obtains precision, recall, and F-measures that fall short of the combination of Ogmios and Caramba by about 1 point (pt) only.

However, most of the observations made on the training set do not carry over to the test set (third set of P-R-F columns: Test, conlleval). 10-fold cross-validation on the training corpus is quite predictive of the results of Caramba alone on the test corpus, with a moderate loss of 2 pt of precision, recall and F-measure. In contrast, the performance of Ogmios on the training corpus was substantially reduced on the test corpus, with a drop of 16 pt in F-measure. Using Ogmios as features in the CRF and training on the training corpus substantially improved its results on the test corpus, bringing them on par with those of Caramba. But adding more features along with Ogmios, including all those of Caramba, did not improve the results on the test corpus: the large improvements obtained when adding a few fea-

Table 5. Caramba: best additional groups of patterns at second iteration (training set).

P	R	F	Description
0.7622	0.6904	0.7245	*Lemma, from TreeTagger
0.7684	0.6851	0.7244	*B: Brown Beth_Partners unigrams
0.7589	0.6898	0.7227	*Normalized token
0.7637	0.6857	0.7226	*Specialist Lexicon syntactic category, with normalized token
0.7624	0.6852	0.7217	B: Specialist Lexicon syntactic category, with normalized token
0.7575	0.6876	0.7209	*TreeTagger POS, with normalized token
0.7595	0.6856	0.7206	B: lemma, from TreeTagger
0.7648	0.6811	0.7205	B: Brown UMLS unigrams
0.7640	0.6796	0.7194	*Section identifier
0.7632	0.6799	0.7192	*Digit
0.7578	0.6837	0.7189	B: Charniak-McClosky POS unigrams, bigrams, trigrams
0.7590	0.6805	0.7176	B: TreeTagger POS, with normalized token
0.7570	0.6819	0.7175	UMLS first or two Semantic Types
0.7487	0.6887	0.7175	*Date
0.7561	0.6821	0.7172	*Alphabetic or case
0.7579	0.6802	0.7169	B: Wmatch
0.7627	0.6757	0.7166	B: section identifier
0.7561	0.6810	0.7166	*B: TreeTagger chunk, BIO
0.7607	0.6772	0.7165	B: Wmatch, BIO
0.7522	0.6775	0.7129	B: date
0.7527	0.7119	0.7317	Subset 2: Subset 1 + all of the above
0.7761	0.6957	0.7337	Subset 3: Subset 1 + starred feature groups only

Notes: B: bigram of classes. Each pattern group is added independently to the pool of Iteration 1 (ie, Subset 1).

Table 6. Combinations of Ogmios (Og) with Caramba (Ca). 10-fold cross validation on the training corpus (except for Ogmios, first row), then application to the test corpus. Pairs of numbers (–n, +m) in the rest of this caption indicate the range of relative positions of n-grams of attributes. All feature sets in the CRF include bigrams of classes (B feature).

	Training						**Test**					
	Conlleval			**i2b2Evaluation**			**Conlleval**			**i2b2Evaluation**		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Og	0.7079	0.8229	0.7611	0.8281	0.9602	0.8893	0.5681	0.6419	*0.6027*	0.7839	0.8852	0.8315
Ca	0.7761	0.6957	*0.7337*	0.9322	0.8336	0.8801	0.7541	0.6787	**0.7144**	0.9210	0.8282	**0.8721**
OgF	0.7581	0.8091	0.7828	0.8648	0.9206	0.8918	0.7469	0.6758	**0.7096**	0.9183	0.8303	**0.8721**
OgT	0.8483	0.8370	**0.8426**	0.9292	0.9144	**0.9217**	0.7443	0.6746	**0.7077**	0.9163	0.8299	**0.8709**
OgCa	0.8613	0.8477	**0.8545**	0.9362	0.9192	**0.9276**	0.7472	0.6795	**0.7117**	0.9159	0.8324	**0.8721**

Notes: Og: Ogmios alone, as is; *Ca*: Caramba alone; OgF: Ogmios output as only attributes: unigrams and bigrams of Ogmios attributes (–1, +1); OgT: Ogmios + normalized token: unigrams and bigrams of Ogmios attributes (–1, +1), with unigrams (–5, +3) and bigrams (–2, +1) of tokens, and one of the previous three tokens; OgCa: Ogmios as feature added to Caramba: unigrams and bigrams of Ogmios attributes (–1, +1), and above subset of Caramba features. Bold shows the (set of) best results per column; italics shows the lowest results when they are notable.

tures to Ogmios on the training set were lost in the test set. In the end, all methods except Ogmios alone (Og) obtained comparable results, within 1 pt of the F-measure. It is notable that wrapping Ogmios as features in the CRF classifier boosted its F-measure by 11 pt on the test set, but adding more features did not gain more than an extra half point.

By design, the i2b2Evaluation measures are much more lenient. As can be expected, the reported scores are well above the strict measures of conlleval, with an increase of 7–15 pt on the training set and 16–23 pt on the test set. They generally kept the order found by the strict measure of conlleval, but tended to reduce the observed differences. This can be interpreted to mean that a large part of the errors made by the systems involve entities which overlap the gold standard entities, but with boundary or type errors (which have little or no impact on the i2b2Evaluation results). The obtained F = 0.8721 would bring either system to the sixth position among 14 participants at the event detection task of the 2012 i2b2/VA challenge, at 4 pt from the top-performing system.

Analysis

Observation of feature usage

The experiments performed so far took the trained classifiers as black boxes. Here we study the contents of the models built and their use in the decisions made by the classifiers. More specifically, we examine which feature contributed most to make a given decision; ie, given that the CRF scores and ranks the possible classes of each token in a sentence, we want to know which feature had the highest weight in putting a class in the top rank instead of the second rank. We make these observations on the test set.

Table 7 lists the top 2 features or feature groups in these decisions. The sum of all Ogmios-based features have the highest score in 80% of the cases. The bigrams of classes feature is often the highest coefficient; it imposes constraints on sequences of classes, eg, I-PROBLEM often follows B-PROBLEM but never follows B-TEST. To put these scores in perspective, we also show the ranges of the total scores for a decision (ie, the sum of the relative values of the scores of the features) and of the total mass of scores (ie, the sum of the absolute values of the scores of the features). This shows that Ogmios features have a very strong weight in the decisions, representing 63% of the final score, and that class bigrams are important to enforce the coherence of the predicted sequence.

The feature with the highest score does not always change the decision that would have been made by the other features. Therefore, we also examined which features most often imposed a decision against the decision that would have been made by the rest of the features.

Table 7. Strongest groups of features to make a decision.

Group of features	**Range of weights**	**Sum of weights**
Ogmios	~[0:8; 4]	~ [2:7; 21:4]
Bigrams of classes feature	~[1:5; 4]	~ [1:5; 4]
Total score		~ [10; 30]
Total mass		Up to ~50

Ogmios imposed the decision in 68% of the cases, with or without the contribution of other features. WMatch (see Table 2), another expert-based set of features, is the 2 most contributing group: it changed the decisions in 8% of the cases, though its effect was far behind that of Ormios. In these cases, Ogmios alone would have made a different decision. WMatch, possibly together with some other features, made the current decision. Among the other feature subsets, the Brown clusters are noticeable in that when Ogmios proposed a wrong decision, the Brown clusters often disagreed. However, their weights were not sufficient to balance those of Ogmios. The B feature (bigrams of classes) also played this role.

Issues with overfitting

The observations made in the previous sections, both when evaluating the combined system as a black box and when studying which features contribute most to decisions, confirm the issue that we had mentioned in our general discussion of system combination methods: the expert-based system overfits the training data.

Wrapping it as features input to the CRF does correct this overfitting significantly, which was not expected. This can be attributed to either or both of 2 factors. 1 is the actual attributes used, which not only include the direct output of Ogmios for the current token as a unigram, but also that for the previous and next tokens, bigrams for the previous and current tokens (these features are computed both for the Ogmios target event types and for the Ogmios contextual tags); and the bigrams of classes (B feature). The second factor is the fact that the CRF computes weights for the feature functions it builds over these Ogmios attributes and the bigrams of classes, thereby possibly giving less confidence to some predicted event types and more to other features such as the bigrams of classes. Knowing which is true will require more detailed investigation.

However, since Ogmios has very good results on the training data, when training the CRF with Ogmios input features and additional features, the CRF trusts the Ogmios features too much and assigns them high scores. But we further observed that Ogmios is not consistent across the training and test data: the errors it makes on the training set are not the same as those it makes on the test set. Because of that, the CRF is unable to learn how to use the other features reliably in a way that will correct the errors that Ogmios makes on the test set.

A method to overcome this issue would be to train the expert-based system and the combining data-driven system on 2 distinct data sets. Ogmios would be developed on a first data set, then its output would be used as input features to train the CRF classifier on a second training set.

In principle, even with an expert-based system, a held-out set (development set) should be kept aside when tuning the system. This held-out set could be used in 2 different ways. One would consist of developing the expert-based system on the training set, and using the union of training and development sets to train the CRF with features obtained from the expert-based system. Another way would also develop the expert-based system on the training set, then analyze its errors on the development set. This could be used to learn a confidence function on the expert-based system output, or to learn how to reproduce its errors. This error model would then be applied to transform expert-based system output on the training set, hence virtually undoing its overfitting on the training set, simulating the errors it is expected to make when applied to other corpora. Unfortunately, in our case, no part of the annotated corpora was left to implement this method.

Conclusion

We presented in this paper experiments on the combination of 2 entity recognition systems: an expert-based system, Ogmios, and a data-driven system, Caramba. By studying the combination of these 2 systems, both as black boxes and in terms of contributing features, we could evidence overfitting of the expert-based system on the training corpus. This made the CRF trust that system too much and prevented it from learning to use other features to correct that system's errors, giving too much weight to the features based on the expert-based system. We highlighted that the use of a development corpus, distinct from the training and test corpora, not only for the data-driven system, but also for the expert-based system, is necessary to prevent this kind of situation.

We also observed that, in contrast, the expert-based system could be substantially improved by simply wrapping it within a linear-chain CRF clas-

sifier, with no other features than the expert-based system's output and the bigram of classes feature. This increased its F-measure by 10 pt from 0.603 to 0.710 as measured by conlleval, due to a boost in precision (+18 pt) and an increase in recall (+3 pt), bringing it on par with the data-driven system (F = 0.714). The increase of F-measure found by the 'overlap' i2b2 evaluation measure is smaller (+4 pt from 0.832 to 0.872) but the conclusions are the same (both CRF-wrapped Ogmios and Caramba reaching for their 0.872 F-measures). We plan to further investigate at the feature level the reasons for this improvement.

Author Contributions

Conceived and designed the experiments: PZ, TL, NG, TH, SR, CG. Analyzed the data: TL, PZ. Wrote the first draft of the manuscript: PZ, TL, NG, TH, SR, CG. Contributed to the writing of the manuscript: PZ, TL, NG, TH, SR, CG. Agree with manuscript results and conclusions: PZ, TL, NG, TH, SR, CG. Jointly developed the structure and arguments for the paper: PZ, TL. Made critical revisions and approved final version: PZ, TL, NG, TH, SR, CG. All authors reviewed and approved of the final manuscript.

Funding

This work has been partly done within the framework of the Quaero project funded by Oseo, French State Agency for Research and Innovation, of the Accordys project, funded by ANR under grant number ANR-12-CORD-0007-03, and of the EDyLex project, funded by ANR under grant number ANR-09-CORD-008-03. The i2b2/VA 2012 challenge and data preparation were supported by Informatics for Integrating Biology and the Bedside (i2b2) award number 2U54LM008748 from the NIH/National Library of Medicine (NLM), by the National Heart, Lung and Blood Institute (NHLBI), and by award number 1R13LM01141101 from the NIH NLM. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NLM, NHLBI, or the National Institutes of Health.

Competing Interests

Authors disclose no potential conflicts of interest.

Disclosures and Ethics

As a requirement of publication the authors have provided signed confirmation of their compliance with ethical and legal obligations including but not limited to compliance with ICMJE authorship and competing interests guidelines, that the article is neither under consideration for publication nor published elsewhere, of their compliance with legal and ethical guidelines concerning human and animal research participants (if applicable), and that permission has been obtained for reproduction of any copyrighted material. This article was subject to blind, independent, expert peer review. The reviewers reported no competing interests.

References

1. Meystre SM, Savova GK, Kipper-Schuler KC, Hurdle JF. Extracting information from textual documents in the electronic health record: a review of recent research. *Yearbook of Medical Informatics*. 2008;35:128–44.
2. Sager N, Friedman C, Lyman MS, editors. *Medical Language Processing: Computer Management of Narrative Data*. Reading, MA: Addison-Wesley, 1987.
3. Friedman C, Alderson PO, Austin JH, Cimino JJ, Johnson SB. A general natural-language text processor for clinical radiology. *J Am Med Inform Assoc*. 1994;1(2):161–74.
4. Hamon T, Grabar N. Linguistic approach for identification of medication names and related information in clinical narratives. *J Am Med Inform Assoc*. 2010;17(5):549–54.
5. Minard AL, Ligozat AL, Ben Abacha A, et al. Hybrid methods for improving information access in clinical documents: concept, assertion, and relation identification. *J Am Med Inform Assoc*. 2011;18(5):588–93.
6. Ralph Grishman and Beth Sundheim. Message Understanding Conference-6: A brief history. In *Proc COLING*. 1996;96:466–71.
7. John Lafferty, Andrew McCallum, Fernando Pereira. Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. *Proc ICML*. 2001:282–9.
8. Sutton C, McCallum A. An introduction to conditional random fields for relational learning. In Getoor L, Taskar B, editor. *Introduction to Statistical Relational Learning*. 1996; Cambridge: MIT Press.
9. Uzuner O, South BR, Shen S, Duvall SL. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *J Am Med Inform Assoc*. 2011;18(5):552–6.
10. Aronson AR, Lang FM. An overview of MetaMap: historical perspective and recent advances. *J Am Med Inform Assoc*. 2010;17(3):229–36.
11. Savova GK, Masanz JJ, Ogren PV, et al. Mayo clinical text analysis and knowledge extraction system (cTAKES): architecture, component evaluation and applications. *J Am Med Inform Assoc*. 2010;17:507–13.
12. Gahbiche-Braham S, Bonneau-Maynard H, Lavergne T, Yvon F. Repérage des entités nommées pour l'arabe : adaptation non-supervisée et combinaison de systèmes. *Actes de la conférence conjointe JEP-TALN-RECITAL*. 2012;2:487–94. http://www.aclweb.org/anthology/F/F12/F12-2044.pdf.
13. Uzuner O, editor. *i2b2/VA 2012 Challenge Workshop*. 2012; Chicago: i2b2.
14. Uzuner O, Bodnari A, Shen S, Forbush T, Pestian J, South BR. Evaluating the state of the art in coreference resolution for electronic medical records. *J Am Med Inform Assoc*. 2012;19(5):786–91.
15. Hamon T, Grabar N. Concurrent linguistic annotations for identifying medication names and the related information in discharge summaries. *Proceedings of the Third i2b2 Workshop on Challenges in Natural Language Processing for Clinical Data*. 2009:1.
16. Tsuruoka Y, Tateishi Y, Kim JD, et al. Developing a robust part-of-speech tagger for biomedical text. *Proceedings of Advances in Informatics—10th Panhellenic Conference on Informatics*. 2005:382–92.

17. Aubin S, Hamon T. Improving term extraction with terminological resources. In *Advances in Natural Language Processing*. 2006;4139:380–7.
18. Olivier Bodenreider. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucl Acids Res*. 2004;32:267–70.
19. *RxNorm Documentation*. National Library of Medicine. 2009. Available at: http://www.nlm.nih.gov/research/umls/rxnorm/docs/index.html. Retrieved May 26, 2013.
20. Lavergne T, Cappé O, Yvon F. Practical very large scale CRFs. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. 2010:504–13.
21. Schmid H. Probabilistic part-of-speech tagging using decision trees. *Proceedings of International Conference on New Methods in Language Processing*. 1994;12:44–9.
22. Galibert O. *Approches et méthodologies pour la réponse automatique à des questions adaptées à un cadre interactif en domaine ouvert*. 2009; Paris: Université Paris-Sud. http://www.sudoc.fr/136783082 and http://tel.archives-ouvertes.fr/tel-00617178/.
23. McClosky D, Charniak E, Johnson M. Automatic domain adapatation for parsing. *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. 2010:28–36.
24. Brown PF, Della Pietra VJ, de Souza PV, Lai JC, Mercer RL. Class-based n-gram models of natural language. *Computational Linguistics*. 1992;18(4):467–79.
25. Percy Liang. Semi-supervised learning for natural language. 2005; Boston: MIT. http://books.google.fr/books/about/Semi_supervised_Learning_for_Natural_Lan.html?id=oNM_NwAACAAJ&redir_esc=y and http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.115.3442.