VUACLTL at SemEval 2016 Task 12: A CRF Pipeline to Clinical TempEval

Tommaso Caselli and **Roser Morante**
CLTL Lab - Vrije Univeristeit Amsterdam
De Boelelaan 1105
1081 HV Amsterdam, The Netherlands
{t.caselli,r.morantevallejo}@vu.nl

Abstract

This paper describes VUACLTL, the system the CLTL Lab submitted to the SemEval 2016 Task Clinical TempEval. The system is based on a purely data-driven approach based on a cascade of seven CRF classifiers which use generic features and little domain knowledge. The challenge consisted in six subtasks related to temporal processing clinical notes from raw text (event and temporal expression detection and attribute classification, temporal relation classification between events and the Document Creation Time, and narrative container detection). The system was initially developed to process newswire texts and then re-trained to process clinical notes. This had an impact on the results, which are not equally competitive for all the subtasks.

1 Introduction

Temporal Processing is becoming more and more important for improving access to content. The availability of timelines (either event-centric or entity-centric) can help improving more complex semantically-focused tasks such as Question Answering, Text Summarization, and Textual Entailment, among others. Furthermore, timelines can be further exploited for monitoring the development in time of different phenomena, e.g. the opinions in debates. Temporal Processing research has mainly focused on the newswire domain (Verhagen et al., 2007; Verhagen et al., 2010; UzZaman et al., 2013) in the framework of several shared tasks where systems were challenged to extract the relevant components of a document timeline: temporal expressions, event mentions, and temporal relations. Several evaluations have shown the capabilities and limits of both the annotated resources and the systems. For instance, the best system in TempEval-3 (Bethard, 2013) reports 0.398 F1 on Temporal Relation Detection and Classification from raw text. The development of temporally annotated corpora has boosted research in languages other than English such as Italian (Caselli et al., 2014), French (Arnulphy et al., 2015), and Spanish (Llorens et al., 2010), among others. Recently, interest in temporal processing has moved forward in two directions: cross-document timeline extraction (Minard et al., 2015) and domain adaptation (Sun et al., 2013; Bethard et al., 2015).

The setting of the 2015 and 2016 SemEval Clinical TempEval Tasks is similar to previous TempEval campaigns, with the two main differences: i.) the domain , i.e. (colon) cancer clinical notes; and ii.) the annotation scheme, i.e. the THYME annotation scheme (Styler IV et al., 2014), an extended version of TimeML (Pustejovsky et al., 2003a). Similarly to the previous edition, the SemEval 2016 Clinical TempEval task (Bethard et al., 2016) consists of the following six subtasks: temporal expression detection (TS) and attribute classification (TA), event detection (ES) and attribute classification (EA), temporal relation detection and classification of an event with respect to the Document Creation Time (DR), and, finally, narrative container relation identification (CR). Systems are evaluated in two phases: Phase 1, which addressed all six subtasks from raw data, and Phase 2, where target entities, such as events and temporal expressions (including their attributes), were given and the systems were evaluated

Proceedings of SemEval-2016, pages 1241–1247,
San Diego, California, June 16-17, 2016.

only against the temporal relation subtasks (DR, and CR). Our team participated in all subtasks and in both submission phases. Our main goals were:

- To test how our full system for temporal processing (from raw text to temporal relations) developed for the newswire domain would perform in another domain using minimal domain specific knowledge, both in terms of lexical resources and tools could achieve a competitive performance;
- To test the robustness of a system that uses simple morpho-syntactic features provided by a standard NLP pipeline(s);

The remainder of the paper is structured as follows: in Section 2, we provide an extensive description of the system and the features we have used.[1] Section 3 reports the results of the submitted runs and their comparison with respect to the baseline system, the median, and the maximum values as provided by the organizers. In Section 4 we perform an error analysis in order to better understand the limits of our system and gain insights for future improvements. Finally, Section 5 puts forward some conclusions.

2 System Description

The task organizers provided 293 training reports, 147 development reports for system development, and 151 testing reports for blind system evaluation. The training and development data had been used in the previous edition of the task.

The general structure of our system can be described as a pipeline of basic NLP tools on top of which we apply several Conditional Random Field (CRF) classifiers (Lafferty et al., 2001). We have used the CRF++ tool with default settings for the regularization algorithm (L2)[2] for all tasks. The final output is obtained by converting the output of 7 different classifiers into the task representation format, i.e. anafora xml files. In the following subsections, we describe the preprocessing steps, which is common for all subtasks, and the specific system for each subtask.

[1]For obtaining scripts and trained models contact the authors.

[2]https://taku910.github.io/crfpp/#links

2.1 Preprocessing

All text files have been preprocessed by using two different tools: the IXA-pipeline (Agerri et al., 2014)[3] and the Stanford CoreNLP tool (Manning et al., 2014). From the IXA pipeline, we used the tokenization, offset and sentence splitting modules. We then passed the tokenized data to the Stanford CoreNLP tool in order to extract additional basic annotation layers such as lemmatization, part-of-speech tagging, and dependency parsing. The preprocessing step outputs the texts in a tab separated column format.

After preprocessing the text files, we merged the preprocessed text with the gold annotations, which were exported from the anafora xml files into a tab-column separated files.

2.2 Span Detection (ES, TS) and Attributes Classification (EA, TA) Tasks

We addressed the ES and TS task as a sequence labeling problem. As for the ES subtask, given an input text, each token is classified as being at the beginning of an event (B-event), inside an event (I-event), or outside an event (O). For this subtask, we have minimally adapted an event classifier developed for the newswire domain (NewsC), by adding domain specific features. We then developed a dedicated classifier for the EA subtasks. The TS and TA tasks have been addressed in a similar way to the ES and EA subtasks though, in this case, the temporal expression detection (TS) and type classification (TA) have been performed in one step by classifying all tokens in a text as being at the beginning or inside of a specific type (B-DATE or B-DURATION, I-DATE, I-DURATION, ...) or outside a temporal expression (O).

The ES and TS/TA subtasks share a set of basic morphosyntactic, namely:

- Token's word, lemma, part-of speech, and dependency relation.
- Full dependency syntax path from the token to the root token.
- A combination of the token's part-of-speech, dependency relation and dependency syntax path to the root token.

[3]https://github.com/ixa-ehu/vmc-from-scratch

The specific features for ES are the following:

- Lemma and part-of-speech of the token's head in the dependency tree.
- Semantic features (PropBank classes, FrameNet frames, and WordNet classes).
- A context window of size +/-2 for word, lemma, and part-of-speech.
- Domain specific feature 1: UMLS entity types [4]. The UMLS types have been assigned by means of a dictionary look-up. The dictionary has been created by means of the manually UMLS annotation from the training and development data.
- Domain specific feature 2: DBPedia "disease" class. Similarly to the UMLS, the DBPedia "disease" class has been assigned by means of a dictionary look-up. The dictionary has been created by extracting all mentions belonging to the class "disease" from DBpedia. [5]

As for the TS/TA specific features we have selected:

- A combination of the token's dependency relation, lemma of its head, and part-of-speech.
- Semantic information (WordNet class and UMLS entity type, only).
- A context window of size +/-5 for token word, lemma, and part-of-speech.
- A context window of size +/-1 with a combination of the token's dependency relation, head's lemma and governor's part-of-speech.

As for the EA subtask, we focused only on the EA:`type`. We have used a reduced set of lexical features with respect to the ES task along with new features from the IXA pipeline, namely:

- The token's word, lemma, and part-of speech;
- A combination of the token's part-of-speech, dependency relation and dependency syntax path to the root.
- Semantic features (PropBank class, FrameNet frame, WordNet class, UMLS entity types and DBpedia "disease").
- The predicate-argument structure from the IXA pipeline.[6]

As for the other EA values, we have assigned to each predicted event the most frequent attribute value as obtained from the training and development data.

2.3 Relation between Event and Document Creation Time Relation (DR)

The DR task was addressed as a multi-class classification task by considering pairs of [*event*, *time*] where each event was paired with the Document Creation Time (DCT). Following the THYME annotation guidelines, we set the DCT to the temporal expression in the first line of the document with the expression "`head start date`". To represent the DCT, we have used only one feature, the predicted class. For each predicted event (as described above), the following features where used :

- The event word, lemma, and part-of-speech;
- The event's dependency relation, the event's head lemma and part-of-speech.
- A combination of the event part-of-speech, dependency relation, and the event's head part-of-speech.
- The predicted class of the event (as described above).
- A context window of +/-2 consisting of lemma, part-of-speech, and whether the token has been predicted either as an event or as a temporal expression.
- Semantic features (PropBank class, FrameNet frame, WordNet class, UMLS entity types, and DBpedia "disease").

2.4 Identifying Narrative Container Relations (CR)

Similarly to the DR task, the CR task was addressed as a classification task involving pairs of [*event*, *event*] and pairs of [*time*, *event*]. We restricted the pairs to intrasentential relations.

[4] `https://www.nlm.nih.gov/research/umls/META3\_current\_semantic\_types.html`

[5] `http://web.informatik.uni-mannheim.de/DBpediaAsTables/DBpediaClasses.htm`

[6] `https://github.com/newsreader/ixa-pipe-srl`

We developed two different approaches. The first approach (CLTLVUA-run1) addresses the problem of CR identification and classification in two steps: first, it automatically identifies candidate events or temporal expressions which can be eligible for being containers, and then it uses this information to create the candidate pairs, i.e. [*event*, *event*] and [*time*, *event*], to detect the presence of a CR relation. On the other hand, the second approach (CLTLVUA-run2) detects and classifies CR relations in a single step. In both approaches the classifiers use the same set of features. The CR detection and classification tasks have been performed with two classifiers: one for [*event*, *event*] pairs and another for [*time*, *event*] pairs.

We used basic morpho-syntactic and semantic features for the container detection model, namely:

- The event/temporal expression's word, lemma and part-of-speech.
- Semantic features (PropBank class, FrameNet frame, WordNet class, UMLS entity types, and DBpedia "disease").
- The temporal expression's class;
- A context window of +/-2 consisting of lemmas and parts-of-speech.
- A combination of the token's part-of-speech, dependency relation and syntactic path to the root.

The CR classifier for [*event*, *event*] pairs uses three sets of features:

- Basic morpho-syntactic and semantic features for each event in the pair (text, lemma, part-of-speech, a combination of part-of-speech, dependency relation and head's part-of-speech, PropBank class, FrameNet frame, WordNet class, UMLS entity types, and DBpedia "disease").
- The syntactic path connecting the two events in the relations (enriched with parts-of-speech).
- Contextual features: temporal prepositions connecting the two events, temporal preposition at the beginning of the sentence and the presence of other events between the element in the pair.

The CR classifier for [*time*, *event*] pairs uses the same set of features as for the [*event*, *event*] classifier plus the temporal expressions class and the textual order of the pair.

3 Results

We report the results on the test set for all subtasks. For clarity's sake we will illustrate in different tables the results for all subtasks. Results have been computed in terms of Precision (P), Recall (R) and F1. For comparison we will also report the baseline provided by the organizers (Bethard et al., 2015), and the median and maximum scores of the participating systems.

Table 1 contains the system scores for ES and EA:`type` for Phase 1 of the evaluation. As for EA:`type` we will report only the F1 score. We also report the results obtained by our system on the newswire domain (test set of the TempEval-3 evaluation).

ES System	P	R	F1	`type`-F1
VUACLTL	0.868	0.828	**0.847**	0.819
Baseline	0.878	0.834	0.855	0.833
Median	0.887	0.846	0.874	0.844
Maximum	0.915	0.891	0.903	0.882
VUACLTL - NewsC	0.861	0.858	0.859	n.a.

Table 1: VUACLTL Results for ES and EA:`type` subtasks - Phase 1.

The results obtained are below the baseline (-0.017 for P and -0.006 for R) and median scores (-0.019 for P and -0.018 for R). In absolute terms, the results are not much different from the NewsC version of the system.

The results for TS and TA are reported in Table 2. We include also an out-of-competition version (VUACLTL_OC), with a bug correction in the conversion script for the final format. The VUACLTL_OC has a lower score for P for the baseline (-0.013) and median (-0.018), while R outperforms baseline and basically equals the median score.

TS System	P	R	F1	`class`-F1
VUACLTL	0.660	0.372	0.476	0.462
VUACLTL_OC	0.761	0.540	**0.632**	0.619
Baseline	0.774	0.428	0.551	0.532
Median	0.779	0.539	0.637	0.618
Maximum	0.840	0.758	0.795	0.772

Table 2: VUACLTL Results for TS and TA subtasks - Phase 1.

Table 3 reports the results for the DR for Phase 1 and Phase 2. For Phase 2 the organizers provided only R scores. In Phase 1 the system scores median results, while in Phase 2 the system R scores above baseline and below median.

DR System	P	R	F1
VUACLTL phase 1	**0.655**	**0.624**	**0.639**
Baseline - phase 1	0.620	0.589	0.604
Median - phase 1	0.655	0.624	0.639
Maximum - phase 1	0.766	0.746	0.756
VUACLTL phase 2	0.724	**0.701**	0.712
Baseline - phase 2	-	0.675	-
Median - phase 2	-	0.724	-
Maximum - phase 2	-	0.843	-

Table 3: VUACLTL Results for DR - Phase 1 and 2.

Finally, Table 4 reports the results for the CR subtask for Phase 1 and 2. In both evaluation phases, both runs of the systems outperform the baseline and median scores for P and R. The VUACLTL_OC also obtains competitive score for P with respect to the maximum score (-0.008). Similar observations hold for Phase 2 of the evaluation where VUACLTL-run1, though performing below the maximum scores, obtains the median scores for R and F1, and a higher P. On the other hand, VUACLTL-run2 tends to maximize R with a minor downgrading of P.

CR System	P	R	F1
VUACLTL-run1 phase 1	**0.497**	0.241	0.325
VUACLTL-run2 phase 1	0.493	**0.268**	**0.347**
VUACLTL_OC-run1 phase 1	**0.523**	0.253	0.341
Baseline - phase 1	0.403	0.067	0.115
Median - phase 1	0.491	0.235	0.318
Maximum - phase 1	0.531	0.471	0.479
VUACLTL-run1 phase 2	**0.642**	0.345	0.449
VUACLTL-run2 phase 2	0.589	**0.368**	**0.453**
Baseline - phase 2	0.459	0.154	0.231
Median - phase 2	0.589	0.345	0.449
Maximum - phase 2	0.823	0.564	0.573

Table 4: VUACLTL Results for CR - Phase 1 and 2.

4 Discussion

Overall, our system obtains competitive scores only in the DR and and CR subtasks while in the other substasks the performances are low.

As for the ES and EA:`type` subtasks, our approach was clearly not the best solution as our system cannot outperform the baseline. We have identified at least three different sources of errors: i) wrong output of the pre-processing modules, especially the tokenization module; ii) limitations of the features selected; and iii) lack of domain specific knowledge (i.e. semantics) and rules to add robustness to the data-driven approach (Valenzuela-Escárcega et al., 2015).

A per-document evaluation of the ES subtask has shown that out of the 151 testing reports half of them have an F1 equal or higher than the median score, 13 have an F1 between the baseline and the median score and 75 have an F1 below the baseline. In this latter group, we have a subset of 7 files with F1 below or equal 0.50. A detailed analysis of these subset has shown that the source of errors (between 46% - 67%) is due to wrong offsets. Different problems, such as lack of domain specific knowledge, errors in parsing[7], and lack of post-processing rules, affect the other 68 files. In particular, an analysis of a subset of the 47 files with P and R below the baseline shows that the false negatives represent between 20% and 37% of the system errors while false positives are only between 8% and 26%. Most of the false negatives are mentions of events that are illnesses (e.g. *tumor*, *adenocarcinoma*) or events with a limited number of annotated examples in the training and development data (e.g. *Grossed* 9 annotated cases out of 32 mentions; *labeled* 15 annotated cases out of 34 mentions). We have also noticed that errors derived from wrong tokenization (and offset) are still present with percentages ranging between 1% to 9%.

Despite the modest performance of the system, it is interesting to observe that features which work for the newswire domain[8] can be easily used to obtain good results also in other domains for this task. It is clear that the results of the ES subtask affects the performance on the EA:`type` subtask. Furthermore, the lack of rules and good domain specific knowledge have also affected the robusteness of the system.

A main factor that affects the performance of the system in the TS and TA subtasks is the choice of tackling span and class identification in one step, instead of two. Nevertheless, the VUACLTL_OC version obtains comparable results for TS for R and F1

[7] For some sentences, the Stanford CoreNLP parser was not able to provide a dependency output.

[8] See the performance of the NewsC system in Table 1.

with respect to the baseline and the median score, but not for P (-0.013 for the baseline, -0.018 for the median) and has a higher F1 score (+0.001 point) for the TA subtask.

In the DR subtask, the system achieved the median score in Phase 1 and obtained a lower R in Phase 2, but in both cases it performs better than baseline. A detailed evaluation by DR type shows that the system performs better for OVERLAP (0.643 F1) and BEFORE (0.736 F1), which is logical, since these types are more frequent than the other two types, AFTER (0.675 F1) and BEFORE/OVERLAP (0.438 F1). The system tends to overpredict BEFORE, which has the highest recall (0.823) while it obtains the highest precision for OVERLAP (0.809). In order to improve the results for this task, different features might be needed related to the section where the event occurs, temporal expressions surrounding the event, and tense and aspect features of the predicates in the event context.

As for the CR subtask, the two versions of our system perform well outperforming both baseline and median scores in both evaluation phases. The low R values are in part due to the fact that we paired *time* and *event* expressions within the same sentence only, ignoring cross-sentence relations. The two-step strategy implemented in VUACLTL-run1 clearly pays in terms of P with a minor impact on R. Notice that the bug in the final format conversion for temporal expressions has an impact also on the overall evaluation of the CRs. The P results of the VUACLTL_OC-run1 version show that the two-step approach scores only -0.008 with respect to the maximum score. Although the difference between the two approaches is not statistically significant ($\chi^2 > 0.05$), the VUACLTL-run1 (and VUACLTL_OC-run1) approach is to be preferred over VUACLTL-run2 because of the way it identifies narrative containers. The method focuses on [*event*, *event*] pairs for CRs in order to narrow down the search of possible pair relations and identify semantic properties of candidate containers. The set of features used to identify CRs is a valid one as the results on Phase 2 show (P and R are higher or equal to the median score for both version of the system).

Looking back at our initial goals, we can conclude that the temporal processing system developed for the newswire domain is portable to the clinical domain, although to achieve a top performance it is necessary to use domain specific tools and lexical resources to improve the feature generation. The system proved to be more robust for the CR and DR tasks, than for the ES and TS tasks.

5 Conclusions

We have described the VUACLTL system for the SemEval-2016 Clinical TempEval. The system is based on a combination of different CRFs classifiers, trained with basic morphosyntactic features and domain specific knowledge. Performances for the basic tasks, although competitive, leave room for improvement. Lack of domain specific knowledge and lack of postprocessing rules have affected the system robustness. However, the system has obtained competitive results for the CR task. Although the performances of the two versions of the system are not statistically significant, we prefer the two-step approach (VUACLTL-run1) because it is more precise and it reflects a more linguistically informed notion of narrative container.

There are many options to improve the system, ranging from fine tuning the pre-processing phase in order to avoid offset misalignments, to the generation of better features for the ES and DR subtasks, or the extension of the CR relations to cross-sentence relations. Very important is to integrate more domain specific knowledge.

As future work, we plan to implement all the improvements mentioned above, and additional improvements that might arise from the in-depth error analysis that we are carrying out in order to gain insight into the limitations of the system and to make informed decisions in the engineering of new features.

Acknowledgments

This work has been supported by EU NewsReader Project (FP7-ICT-2011-8 grant 316404), the NWO Spinoza Prize project Understanding Language by Machines (sub-track 3).

References

Rodrigo Agerri, Josu Bermudez, and German Rigau. 2014. Ixa pipeline: Efficient and ready to use multi-

lingual nlp tools. In *LREC*, volume 2014, pages 3823–3828.

Béatrice Arnulphy, Vincent Claveau, Xavier Tannier, and Anne Vilnat. 2015. Supervised machine learning techniques to detect timeml events in french and english. In *Natural Language Processing and Information Systems*, pages 19–32. Springer.

Steven Bethard, Leon Derczynski, Guergana Savova, Guergana Savova, James Pustejovsky, and Marc Verhagen. 2015. Semeval-2015 task 6: Clinical tempeval. *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*.

Steven Bethard, Guergana Savova, Wei-Te Chen, Leon Derczynski, James Pustejovsky, and Marc Verhagen. 2016. Semeval-2016 task 12: Clinical tempeval. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, California, June. Association for Computational Linguistics.

Steven Bethard. 2013. Cleartk-timeml: A minimalist approach to tempeval 2013. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 2, pages 10–14.

Tommaso Caselli, Rachele Sprugnoli, Manuela Speranza, and Monica Monachini. 2014. EVENTI: EValuation of Events and Temporal INformation at Evalita 2014. In *Proceedings of the First Italian Conference on Computational Linguistics CLiC-it 2014 & and of the Fourth International Workshop EVALITA 2014*, pages 27–34. Pisa University Press.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th ICML*, pages 282–289.

Hector Llorens, Estela Saquete, and Borja Navarro. 2010. Tipsem (english and spanish): Evaluating crfs and semantic roles in tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 284–291. Association for Computational Linguistics.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. System Demonstrations*, pages 55–60.

Anne-Lyse Minard, Manuela Speranza, Eneko Agirre, Itziar Aldabe, Marieke van Erp, Bernardo Magnini, German Rigau, Ruben Urizar, and Fondazione Bruno Kessler. 2015. Semeval-2015 task 4: Timeline: Cross-document event ordering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 778–786.

James Pustejovsky, José Castao, Robert Ingria, Roser Saurì, Robert Gaizauskas, Andrea Setzer, and Graham Katz. 2003a. TimeML: Robust Specification of Event and Temporal Expressions in Text. In *Fifth International Workshop on Computational Semantics (IWCS-5)*.

William F Styler IV, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C de Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, et al. 2014. Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143–154.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association*, 20(5).

Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. 2013. Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Marco A Valenzuela-Escárcega, Gus Hahn-Powell, Thomas Hicks, and Mihai Surdeanu. 2015. A domain-independent rule-based framework for event extraction. In *Association for Computational Linguistics (ACL)*.

Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. 2007. Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 75–80. Association for Computational Linguistics.

Marc Verhagen, Roser Sauri, Tommaso Caselli, and James Pustejovsky. 2010. Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 57–62. Association for Computational Linguistics.