SemEval-2015 Task 5: QA TEMPEVAL - Evaluating Temporal Information Understanding with Question Answering

Hector Llorens[♣], Nathanael Chambers[♢], Naushad UzZaman[♣], Nasrin Mostafazadeh[⋈], James Allen[⋈], James Pustejovsky[♠]

[♣] Nuance Communications, USA

[♢] United States Naval Academy, USA

[⋈] University of Rochester, USA

[♠] Brandeis University, USA

hector.llorens@nuance.com, nchamber@usna.edu

Abstract

QA TempEval shifts the goal of previous TempEvals away from an intrinsic evaluation methodology toward a more extrinsic goal of question answering. This evaluation requires systems to capture temporal information relevant to perform an end-user task, as opposed to corpus-based evaluation where all temporal information is equally important. Evaluation results show that the best automated TimeML annotations reach over 30% recall on questions with 'yes' answer and about 50% on easier questions with 'no' answers. Features that helped achieve better results are event coreference and a time expression reasoner.

1 Introduction

QA TempEval is a follow up of the TempEval series in SemEval: TempEval-1 (Verhagen et al., 2007), TempEval-2 (Verhagen et al., 2010), and TempEval-3 (UzZaman et al., 2013). TempEval focuses on evaluating systems that extract temporal expressions (timexes), events, and temporal relations as defined in the TimeML standard (Pustejovsky et al., 2003) (timeml.org). QA TempEval is unique in its focus on evaluating temporal information that directly address a QA task. TimeML was originally developed to support research in complex temporal QA within the field of artificial intelligence (AI). However, despite its original goal, the complexity of temporal QA has caused most research on automatic TimeML systems to focus on a more straightforward temporal information extraction (IE) task. QA TempEval still requires systems to extract temporal relations just like previous TempEvals, however, the QA evaluation is solely based on how well the relations answer questions about the documents. It is no longer about annotation accuracy, but rather the accuracy for targeted questions.

Not only does QA represent a more natural way to evaluate temporal information understanding (UzZaman et al., 2012), but also annotating documents with question sets requires much less expertise and effort for humans than corpus-based evaluation which requires full manual annotation of temporal information. In QA TempEval a document does not require the markup of all the temporal entities and relations, but rather a markup of a few key relations central to the text. Although the evaluation schema changes in QA TempEval, the task for participating systems remains the same: extracting temporal information from plain text documents.

Here we re-use TempEval-3 task ABC, where systems are required to perform end-to-end TimeML annotation from plain text, including the complete set of temporal relations (Allen, 1983). However, unlike TempEval-3, there are no subtasks focusing on specific elements (such as an event extraction evaluation). Also, instead of IE performance measurement for evaluation, a QA performance (on a set of human-created temporal questions on documents) is used to rank systems. The participating systems are supposed to annotate temporal entities relations across the document, and the relations are used to build a larger knowledge base of temporal links to obtain answers to the temporal questions.

In QA TempEval, annotators are not required to tag and order all events, but instead ask questions about temporal relations that are relevant or interesting to the document, hence this evaluation bet-

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 792–800,
Denver, Colorado, June 4-5, 2015.

ter captures the understanding of the most important temporal information in a document. Annotators are not limited to relations between entities appearing in the same or consecutive sentences, i.e., they can ask any question that comes naturally to a reader's mind, e.g., "did the election happen (e3) before the president gave (e27) the speech". Finally, QA TempEval is unique in expanding beyond the news genre and including Wikipedia articles and blog posts. In the upcoming sections we will discuss details of the conducted task and evaluation methodology.

2 Task Description

The task for participant systems is equivalent to TempEval-3 task ABC, see Figure 1. Systems must annotate temporal expressions, events, and temporal relations between them[1]. The input to participants is a set of unannotated text documents in TempEval-3 format. Participating systems are required to annotate the plain documents following the TimeML scheme, divided into two types of elements:

- **Temporal entities**: These include **events** (EVENT tag, "came", "attack") and temporal expressions (**timexes**, TIMEX3 tag, e.g., "yesterday", "8 p.m.") as well as their attributes like event class, timex type, and normalized values.

- **Temporal relations**: A temporal relation (tlink, TLINK tag) describes a pair of entities and the temporal relation between them. The TimeML relations map to the 13 Allen interval relations. The included relations are: SIMULTANEOUS (and IDENTITY), BEFORE, AFTER, IBEFORE, IAFTER, IS_INCLUDED, INCLUDES (and DURING), BEGINS, BEGUN_BY, ENDS, and ENDED_BY. Since the TimeML DURING does not have a clear mapping, we map it to SIMULTANEOUS for simplicity. The following illustrates how the expression "6:00 pm" BEGINS the state of being "in the gym".

  (1) John was in the gym between 6:00 p.m and 7:00 p.m.

Each system's annotations represent its temporal knowledge of the documents. These annotations are then used as input to a temporal QA system (UzZaman et al., 2012) that will answer questions on behalf of the systems, and the accuracy of their answers is compared across systems.

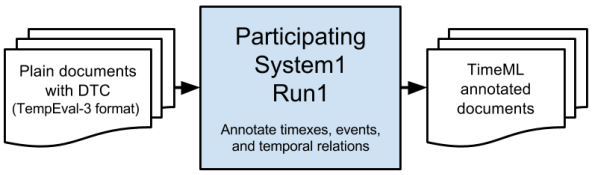

Figure 1: Task - Equivalent to TempEval-3 task ABC

3 QA Evaluation Methodology

The main difference between QA TempEval and earlier TempEval editions is that the systems are not scored regarding how similar their annotation to a human annotated key is, but how useful is their TimeML annotation to answer human annotated temporal questions. There are different kinds of temporal questions that could be answered given a TimeML annotation of a document. However, this first QA TempEval focuses on yes/no questions in the following format:

```
IS <entityA> <RELATION> <entityB> ?
(e.g., is event-A before event-B ?)
```

This makes it easier for human annotators to create accurate question sets with their answers. Other types of questions such as list-based make it more difficult and arguable in edge cases (e.g., list events between event-A and event-B). Questions about events not included in the document are not possible, but theoretically one could ask about any time reference. Due to the difficulty of mapping external time references to a specific time expression in the document, these types of questions are not included in the evaluation.

The questions can involve any of the thirteen relations described above. Two relations not in the set of thirteen, OVERLAPS and OVERLAPPED_BY, cannot be explicitly annotated in TimeML, but they could happen implicitly (i.e., be inferred from other relations) if needed by an application.

The evaluation process is illustrated in Figure 2. After the testing period, the participants send their TimeML annotations of the test documents. Organizers evaluate the TimeML annotations of all the participating systems with a set of questions. The

[1] http://alt.qcri.org/semeval2015/task5

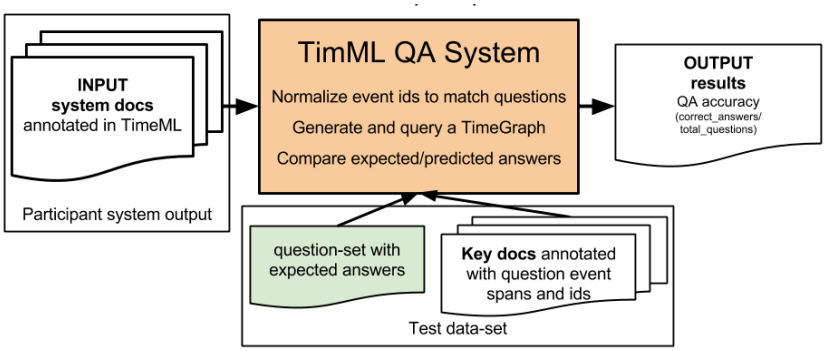

Figure 2: QA based on participant annotations

systems are scored comparing the expected answers provided by human annotators against the predicted answers obtained from the system's TimeML annotations.

Given a system's TimeML annotated documents, the process consists of three main steps:

- **ID Normalization**: Entities are referenced by TimeML tag ids (e.g., eid23). The yes/no questions must contain two entities with IDs (e.g., "is event[eid23] after event[eid99] ?"). The entities of the question are annotated in the corresponding key document. However, systems may provide different ids to the same entities. Therefore, we align the system annotation IDs with the question IDs that are annotated in the key docs using the TempEval-3 normalization tool[2].

- **Timegraph Generation**: The normalized TimeML docs are used to build a graph of time points representing the temporal relations of the events and timexes identified by each system. Here we use Timegraph (Gerevini et al., 1993) for computing temporal closure as proposed by Miller and Schubert (1990). The Timegraph is first initialized by adding the TimeML explicit relations. Then the Timegraph's reasoning mechanism infers implicit relations through rules such as transitivity. For example, if eventA BEFORE eventB and eventB BEFORE eventC, then the implicit relation eventA BEFORE eventC can be inferred. Timegraph expands a system's TimeML annotations and can answer both explicit and implicit Allen temporal relation questions, including OVERLAPS.

- **Question Processing**: Answering questions requires temporal information understanding and reasoning. Note that asking 'IS <entity1> <relation> <entity2>?' is not only asking if there is that explicit tlink between them, but also, if it is not, if that relation can be inferred from other tlinks implicitly. Unlike corpus based evaluation, the system gets credit if its annotations provide the correct answer regardless of whether it annotates other irrelevant information or not. In order to answer the questions about TimeML entities (based on time intervals) using Timegraph, we convert the queries to point-based queries. For answering yes/no questions, we check the necessary point relations in Timegraph to verify an interval relation. For example, to answer the question "*is event1* AFTER *event2*", our system verifies whether start(event1) > end(event2); if it is verified then the answer is true (YES), if it conflicts with the Timegraph then it is false (NO), otherwise it is UNKNOWN.

4 QA Scoring

For each question we compare the obtained answer from the Timegraph (created with system annotations) and the expected answer (human annotated). The scoring is based on the following Algorithm 1. With this we calculate the following measures:

- **Precision (P)** $= \frac{num\_correct}{num\_answered}$
- **Recall (R)** $= \frac{num\_correct}{num\_questions}$

[2]https://github.com/hllorens/timeml-normalizer

```
num_questions=0
num_answered=0
num_correct=0
foreach question q ∈ questionset do
    num_questions += 1
    if predicted_ans[q] != unknown
    or key_ans[q] == unknown then
        num_answered += 1
        if predicted_ans[q] == key_ans[q] then
            num_correct += 1
```

Algorithm 1: QA Scoring

- **F-measure (F1)** $= \frac{2*P*R}{P+R}$

We use Recall (QA accuracy) as the main metric and F1 is used in case of draw.

5 Datasets

In QA TempEval, the creation of datasets does not require the manual annotation of all TimeML elements in source docs. The annotation task in QA TempEval only requires reading the doc, making temporal questions, providing the correct answers, and identifying entities included in the questions by bounding them in the text and designating an ID. The format of the question sets is as follows:

```
<question-num>|<source-doc>|
<question-with-ids>|<NL-question>|
<answer>|[opt-extra-info]
```

Following is an example question and its corresponding annotated document:

```
3|APW.tml|IS ei21 AFTER ei19|
Was he cited after becoming general?|yes

APW.tml (KEY)
Farkas <event eid="e19">became</event>
a general. He was
<event eid="e21">cited</event>...

APW.tml (system annotation, full-TimeML)
Farkas <event eid="e15"...>became</event>
a general. He was
<event eid="e24"...>cited</event>...
<tlink eventID=e15 relatedToEventID=e24
relType=before />
```

5.1 Training Data

TimeML training data consists of TempEval-3 annotated data: TimeBank, AQUAINT (TBAQ, TempEval-3 training), and TE-3 Platinum (TempEval-3 testing). Furthermore, a question-set in the format explained earlier is provided to the participants for training purposes. It consists of 79 Yes/No questions and answers about TimeBank documents (UzZaman et al., 2012). Participants can easily extend the question-set by designing new questions over TimeML corpora.

5.2 Test Data

The test dataset comprises three domains:

- News articles (Wikinews, WSJ, NYT): This covers the traditional TempEval domain used in all the previous editions.
- Wikipedia[3] articles (history, biographical): This covers documents about people or history, which are rich in temporal entities.
- Informal blog posts (narrative): We hand selected blog entries from the Blog Authorship Corpus (Schler et al., 2006). They are in narrative nature, such as the ones describing personal events as opposed to entries with opinions and political commentary.

For each of these domains, human experts select the documents, create the set of questions together with the correct answer, and annotate the corresponding entities of the questions in the key documents. The resulting question-set is then peer-reviewed by the human experts. Table 1 depicts statistics of the test dataset. In this table, the column *dist-* shows the number of questions about entities that are in the same or consecutive sentences while *dist+* refers to questions about non-consecutive (more distant) entities.

	docs	**words**	**quest**	**yes**	**no**	**dist-**	**dist+**
news	10	6920	99	93	6	40	59
wiki	10	14842	130	117	13	58	72
blogs	8	2053	65	65	0	30	35
total	28	23815	294	275	19	128	166

Table 1: Test Data

Annotators were asked to create positive (yes) questions unless a negative (no) question came naturally. This is due to the fact that we can automatically generate negative questions from positive questions, but not the other way around. Note that the number of questions about distant entities is considerable. TimeML training data and thus systems tend to only annotate temporal relations about less

[3] http://en.wikipedia.org

distant entities. Therefore, to answer distant questions the necessary implicit relations must be obtainable from the annotated explicit relations.

5.3 Development Time Cost

One of the claims of QA evaluation of temporal text understanding (UzZaman et al., 2012) is that the time cost of creating question sets in QA schema is lower than the one for fully annotating a document with TimeML elements and attributes. Both tasks involve reading the document. However, question-set creation only requires designing yes/no questions paired with answers and annotating the corresponding entities in the document, while full TimeML annotation needs identifying all entities, their attributes, and large set of relations among them. There is not any rigorous information available about the time it takes to perform these different annotation tasks. Comparison is difficult since many factors play a role in timing (e.g., human annotators skills, dedicated software help). In order to provide an approximate comparison, following we present information regarding some real experiences:

- Question Set annotation (about 10 questions per document, without dedicated software help): QA TempEval consists of 28 docs (23,815 words), i.e., about 850 words per document. Human annotators reported that the annotation task from raw text took them 30min-2h per document, i.e., 15min-1h for 360 words.
- TimeML all-elements and attributes annotation (with dedicated software help): Annotators of the Spanish TimeBank spent a year to complete the annotation working 3h/day, approximately 3h per document or 360 words. We don't have available to us similar data for the English TimeBank's creation.
- Other experiences regarding full TimeML annotation such as correcting a pre-annotated document by a system took about 2-3h per document. TLINK annotation reportedly took about 1.5h per document.

We do not aim to provide an exact quantification or comparison; however, based on the information we have available, creating a QA test set takes considerably less time than full TimeML annotation. TimeML annotated documents can also be used for training and evaluating temporal extraction systems, whereas TempQA annotated documents can be used only for evaluation. Given that we have enough annotated data, TempQA helps to easily create more data to evaluate temporal systems in new domains.

6 Participating Systems

Nine approaches addressing automatic TimeML annotation for English were presented in the QA TempEval evaluation, divided into two groups:

Regular participants, optimized for task:

- **HITSZ-ICRC**[4]. rule-based timex module, SVM (liblinear) for event and relation detection and classification
- **hlt-fbk-ev1-trel1**. SVM, separated event detection and classification, without event coreference
- **hlt-fbk-ev1-trel2**. SVM, separated event detection and classification, with event coref
- **hlt-fbk-ev2-trel1**. SVM, all predicates are events and classification decides, without event co-reference
- **hlt-fbk-ev2-trel2**. SVM, all predicates are events and classification decides, with event coreference

Off-the-Shelf Systems, not optimized on task:

- **CAEVO**[5] (Chambers et al., 2014). Cascading classifiers that add temporal links with transitive expansion. A wide range of rule-based and supervised classifiers are included
- **ClearTK**[6] (Bethard, 2013) A pipeline of machine-learning classification models, each of which have simple morphosyntactic annotation pipeline as feature set
- **TIPSemB** (Llorens et al., 2010) CRF-SVM model with morphosyntactic features
- **TIPSem** (Llorens et al., 2010) TIPSemB + lexical (WordNet) and combinational (PropBank roles) semantic features

[4] Annotations Submitted 1-day after the deadline
[5] Off-the-shelf system: the author was co-organizer
[6] Off-the-shelf system: trained and tested by organizers

7 Time Expression Reasoner (TREFL)

As an extra evaluation, task organizers added a new run for each system augmented with a post-processing step. The goal is to analyze how a general time expression reasoner could improve results. The TREFL component is straightforward: resolve all time expressions, and add temporal relations between the time expressions when the relation is unambiguous based on their resolved times.

We define a "timex reference" as a temporal expression consisting of a date or time (e.g., "Jan 12, 1999", "tomorrow") that is normalized to a Gregorian calendar interval (e.g., 1999-01-12, 2015-06-06). These are perfectly suited for ordering in time. In addition, finding timex references and obtaining their normalized values are tasks in which automatic systems perform with over 90% accuracy. Thus, given a system normalized-values, we can automatically produce timex-timex reference relations or links (**TREFL**) that represent a temporal relation backbone (base Timegraph) with high accuracy. This backbone can then assist the much more difficult event-event and event-timex links that are later predicted by system classifiers. Any relations predicted by a classifier can be discarded if they are inconsistent with this TREFL backbone.

For example, if a system TimeML annotation contains three timexes **t1** (1999), **t2** (1998-01-15), and **t3** (1999-08), a minimal set of relations can be deterministically extracted as *t2* BEFORE *t1* and *t3* IS_INCLUDED *t1*. The corresponding Timegraph is:

t2 $<$ **t1_start** $<$ **t3_start** $<$ **t3_end** $<$ **t1_end**

To automatically obtain such minimal set of relations from the system timex-values, the TREFL component orders them by date and granularity using SIMULTANEOUS, BEFORE, BEGINS, IS_INCLUDED, or ENDS relations. More complicated cases have not been included in this evaluation for simplicity.

The only drawback or risk of this strategy is that some of the system timex-values could be incorrect, but previous work suggests these errors are less numerous than those occurring in later event-event relation extraction. Our hypothesis is that (i) many systems do not include a strategy like this, and (ii) even taking into account the drawback of this strategy most systems would benefit from using it, reaching a higher performance. The evaluation compares original systems with their TREFL-augmented variant that discarded system relations in conflict with its TREFL Timegraph.

8 Evaluation

The objective of this evaluation is to measure and compare QA performance of TimeML annotations of participating and off-the-shelf systems. Participants were given the documents of the previously defined test set (TE3-input format). They were asked to annotate them with their systems within a 5-day period. Organizers evaluated the submitted annotations using the test question-sets. Result tables include Precision (P), Recall (R), F-measure (F1), percentage of the answered questions (awd%) and number of correct answers (corr). As mentioned earlier, Recall is the main measure for ranking systems. The percentage of the questions which are answered by the system provides a coverage metric, measuring a system's ability to provide more complete set of annotation on entities and relations.

8.1 Results without TREFL

Table 2 shows the combined results over all three genres in the test set, comprising 294 test questions.

	Measures			Questions	
System	P	R	F1	awd%	corr
HITSZ-ICRC	.54	.06	.12	.12	19
hlt-fbk-ev1-trel1	**.57**	.17	.26	.30	50
hlt-fbk-ev1-trel2	.47	.23	.31	.50	69
hlt-fbk-ev2-trel1	.55	.17	.26	.32	51
hlt-fbk-ev2-trel2	.49	**.30**	**.37**	**.62**	**89**
ClearTK	.59	.06	.11	.10	17
CAEVO	.56	**.17**	**.26**	**.31**	**51**
TIPSemB	.47	.13	.20	.28	38
TIPSem	**.60**	.15	.24	.26	45

Table 2: QA Results over all domains.

The participant system hlt-fbk-ev2-trel2 system (.30 R) outperformed all the others by a significant margin. CAEVO performed best among the off-the-shelf systems, but behind the winning participant recall by 13% absolute. The awd% of the hlt-fbk-ev2-trel2 system doubles the one by the best off-the-shelf system, CAEVO. Interstingly, CAEVO and the two hlt-fbk *trel1* systems performed approximately the same. The *trel2* versions included event coreference.

Table 3 shows three result tables from the three genres: news, wiki, and blogs. The best overall sys-

News Genre Results

	Measures			Questions	
System	P	R	F1	awd%	corr
HITSZ-ICRC	.47	.08	.14	.17	8
hlt-fbk-ev1-trel1	**.59**	.17	.27	.29	17
hlt-fbk-ev1-trel2	.43	.23	.30	.55	23
hlt-fbk-ev2-trel1	.56	.20	.30	.36	20
hlt-fbk-ev2-trel2	.43	**.29**	**.35**	**.69**	**29**
ClearTK	**.60**	.06	.11	.10	6
CAEVO	.59	**.17**	**.27**	.29	**17**
TIPSemB	.50	.16	.24	**.32**	16
TIPSem	.52	.11	.18	.21	11

Wiki Genre Results

	Measures			Questions	
System	P	R	F1	awd%	corr
HITSZ-ICRC	**.83**	.08	.14	.09	10
hlt-fbk-ev1-trel1	.55	.16	.25	.29	21
hlt-fbk-ev1-trel2	.52	.26	.35	.50	34
hlt-fbk-ev2-trel1	.58	.17	.26	.29	22
hlt-fbk-ev2-trel2	.62	**.36**	**.46**	**.58**	**47**
ClearTK	.60	.05	.09	.08	6
CAEVO	.59	.17	.26	**.28**	22
TIPSemB	.52	.13	.21	.25	17
TIPSem	**.74**	**.19**	**.30**	.26	**25**

Blogs Genre Results

	Measures			Questions	
System	P	R	F1	awd%	corr
HITSZ-ICRC	.17	.02	.03	.09	1
hlt-fbk-ev1-trel1	**.57**	.18	.28	.32	12
hlt-fbk-ev1-trel2	.43	.18	**.26**	.43	12
hlt-fbk-ev2-trel1	.47	.14	.21	.29	9
hlt-fbk-ev2-trel2	.34	**.20**	.25	**.58**	**13**
ClearTK	**.56**	.08	.14	.14	5
CAEVO	.48	**.18**	**.27**	**.38**	**12**
TIPSemB	.31	.08	.12	.25	5
TIPSem	.45	.14	.21	.31	9

Table 3: QA Results broken down by genre, based on 99 News, 130 Wiki, and 65 Blog questions.

tem, hlt-fbk-ev2-trel2, maintained its top position.

The main difference in genre results appears to be the smaller blog corpus where the leading hlt-fbk-ev2-trel2 participant and CAEVO performed similarly, .20 and .18 R respectively. The hlt-fbk system exhibited similar behavior as the other genres showing a high coverage, as demonstrated by awd% metric. However, it simply guessed incorrectly much more often (precision dropped to the 30's).

We make note that the ClearTK off-the-shelf system's lower performance is because it was used without modification from its TempEval-3 submission. ClearTK was TempEval-3 best system, partly due to its optimization to the task where it maximized precision and not recall. It likely would perform better if optimized to this new QA task.

8.2 Results with TREFL

Table 4 shows the results for systems augmented with TREFL (explained in Section 7).

	Measures			Questions	
System	P	R	F1	awd%	corr
HITSZ-ICRC	.58	.09	.15	.15	25
hlt-fbk-ev1-trel1	**.62**	.28	.38	.45	81
hlt-fbk-ev1-trel2	.55	.31	**.40**	.57	92
hlt-fbk-ev2-trel1	.61	.29	.39	.48	86
hlt-fbk-ev2-trel2	.51	**.34**	**.40**	**.67**	**99**
ClearTK (TREFL not applied because of its TLINK format)					
CAEVO	.60	.21	.32	.36	63
TIPSemB	.64	.24	.35	.37	70
TIPSem	**.68**	**.27**	**.38**	**.40**	**79**

Table 4: QA Results **augmented with TREFL**

Recall went up on all systems (by 49% relative on average), but the degree of improvement varied. Recall of the top system (hlt-fbk-ev2-trel2) improved 4% absolute (13% relative). The largest gain was with TIPSem which improved from .15 to .27, becoming the top off-the-shelf system. TREFL is mainly focused on improving recall which explains the differences. The best system had higher recall already, so TREFL had less contribution. TIPSem had lower recall, so it sees the greatest gain. TREFL did not penalize TIPSem precision as much as it did for other systems. That made TIPSem obtain the top F1 in wiki and blogs domains.

By genre, on average TREFL improved systems' *relative* recall by 60% (news), 48% (wiki), and 47% (news).

In news and wiki, hlt-fbk-ev2-trel2+terfl was the system answering correctly more questions about distant entities (22 news, 20 wiki), while for blogs it was TIPSem (9).

We also found that hlt-fbk-ev2-trel2+terfl answers more questions that no other system is capable of answering (4 news, 11 wiki, 5 blogs), demonstrating that it has some features that others system lack. One of the distinguishing features of this system, required to answer some of the testset questions, is event co-reference (clustering) which could be responsible for this good result.

Analyzing the questions answered correctly after the TREFL augmentation, in both the news and wiki domains, we found that around 35% of the questions were not answered by any system because they didn't find a temporal entity in the question (either an event or time expression, or both). This is mostly because no system found one of the entities in the question. In the blog genre, 50% of errors were due to missing entities, and blogs/news were 75%. These missing entity errors exist in both the original system submissions and this TREFL augmentation. The remaining unanswered questions were simply due to sparsity in relation annotation. The relation needed to answer the question is neither annotated nor do transitive inferences exist.

8.3 Results with TREFL (no-questions)

As mentioned earlier the evaluation is mainly focused on positive questions (with *yes* answer) since annotating them provides more information and negative questions can be automatically generated from them. Moreover, in general, answering positive questions is more challenging, e.g., asking IS e1 BEFORE e2 requires a system to guess the single correct relation if the correct answer is *yes*; However, if the correct answer is *no*, there are 12 possible correct relations (all but BEFORE).

In order to have more insight into this issue, we automatically obtained negative questions by asking about the opposite[7] relation with "no" as the expected answer. For example, IS e1 BEFORE e2 (yes) becomes IS e1 AFTER e2 (no). The aim of this evaluation is to analyze system performance in determining if a relation is not correct. In this easier test, participating and off-the-shelf systems obtain better results going over .50 R in the news domain. The best obtained recalls are .52 in news, .39 in wiki, and .42 in blogs, as compared to .38, .36 and .22 obtained for *yes*-questions in the main test.

It is interesting to see that in this negative alternative, systems were better in blogs than in wiki, unlike in the positive test. Likewise the positive variant, the addition of trefl has improved results, but the improvements is smaller in this case.

[7] SIMULTANEOUS has no opposite and IAFTER was used.

9 Conclusions and Future Work

QA evaluation task attempts to measure how far we are on temporal information understanding applied to temporal QA (an extrinsic task) instead of only TimeML annotation accuracy. One of the benefits of QA evaluation is that test set creation time and human expertise required is considerably less than in TimeML annotation. QA TempEval also included Wikipedia and blog domains, in addition to the regular news domain, for the first time. Evaluation results suggest that we are still far from systems that more deeply understand temporal aspects of natural language and can answer temporal questions. The best overall recall was 30% (34% with TREFL). This top result is higher than best off-the-shelf system 17% (27% with TREFL).

The main findings include:

- The only system using event co-reference obtained the best results, so adding event coref may help other systems.
- Adding TREFL improved the QA recall of all systems, ranging from 3% to 12% absolute (13% to 80% relative).
- Training data is news, but the best system performed well on Wikipedia. Some off-the-shelf systems even performed better on Wikipedia/blogs than on the news domain.
- Human annotators annotated as many questions about close entities as distant entities. In the same line, automated systems were capable of answering correctly approximately the same amount of questions of each type.

As future work we aim to extend the analysis of the results presented in this paper. On the one hand, by explaining TREFL technique and its effects in more detail. On the other hand, by finding out what features made some systems unique being the only ones capable of answering certain questions correctly. The question-sets[8], tools and results[9] have been released for future research.

Acknowledgments

We want to thank participant Paramita Mirza for her collaboration on reviewing and correcting the test data, and also Marc Verhagen and Roser Sauri for their help on approximating the time-cost of human TimeML annotation.

[8] http://bitbucket.org/hector_llorens/qa-tempeval-test-data

[9] http://alt.qcri.org/semeval2015/task5/

References

James F. Allen. 1983. Maintaining knowledge about temporal intervals. *Communications of ACM*, 26(11):832–843.

Steven Bethard. 2013. Cleartk-timeml: A minimalist approach to tempeval 2013. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 10–14.

Nathanael Chambers, Taylor Cassidy, Bill McDowell, and Steven Bethard. 2014. Dense event ordering with a multi-pass architecture. *Transactions of the Association for Computational Linguistics*, 2(10):273–284.

Alfonso Gerevini, Lenhart Schubert, and Stephanie Schaeffer. 1993. Temporal reasoning in Timegraph I–II. *SIGART Bulletin*, 4(3):21–25.

Hector Llorens, Estela Saquete, and Borja Navarro. 2010. TIPSem (English and Spanish): Evaluating CRFs and Semantic Roles in TempEval-2. In *Proceedings of SemEval-5*, pages 284–291.

Stephanie Miller and Lenhart Schubert. 1990. Time revisited. In *Computational Intelligence*, volume 26, pages 108–118.

James Pustejovsky, José M. Castaño, Robert Ingria, Roser Saurí, Robert Gaizauskas, Andrea Setzer, and Graham Katz. 2003. TimeML: Robust Specification of Event and Timexes in Text. In *IWCS-5*.

Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W. Pennebaker. 2006. Effects of age and gender on blogging. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, pages 199–205.

Naushad UzZaman, Hector Llorens, and James Allen. 2012. Evaluating temporal information understanding with temporal question answering. In *Proceedings of IEEE International Conference on Semantic Computing*.

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 *2nd Joint Conference on Lexical and Computational Semantics (*SEM), Vol 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, Atlanta, Georgia, USA, June.

Marc Verhagen, Robert Gaizauskas, Mark Hepple, Frank Schilder, 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, Prague.

Marc Verhagen, Roser Saurí, Tommaso Caselli, and James Pustejovsky. 2010. Semeval-2010 task 13: Tempeval-2. In *Proceedings of SemEval-5*, pages 57–62.