Analysis and Refinement of Temporal Relation Aggregation

Taylor Cassidy
IBM Research
Army Research Laboratory
Adelphi, MD 20783, USA
taylor.cassidy.ctr@mail.mil

Heng Ji
Computer Science Department
Rensselaer Polytechnic Institute
Troy, NY 12180, USA
jih@rpi.edu

Abstract

To obtain a complete temporal picture of a relation it is necessary to aggregate fragments of temporal information across relation instances in text. This process is non-trivial even for humans because temporal information can be imprecise and inconsistent, and systems face the additional challenge that each of their classifications is potentially false. Even a small amount of incorrect proposed temporal information about a relation can severely affect the resulting aggregate temporal knowledge. We motivate and evaluate three methods to modify temporal relation information prior to aggregation to address this challenge.

1 Introduction

Temporal information about relations is conveyed in text at varying levels of completeness and specificity. A sentence may indicate that a relation starts, ends, or that it is ongoing at a particular time. Furthermore, a time expression may be expressed at a variety of granularity levels (e.g., hour, day, or year). For instance, *"Collins, ..., is a 61-year-old veteran who went 444-434 in six seasons as a manager, 1994-1996 with Houston"* provides bounds on both the start and end date of the a relation but at a coarse granularity. Conversely, *"Ivory Coast President Laurent Gbagbo on state television Friday dissolved parliament"* conveys temporal information about an arbitrary part of Gbagbo's presidency at a finer granularity: the relation simply holds true at the document creation time (DCT). Single instances in which a relation of interest is related to a time expression often fail to convey complete, fine-grained temporal information. Thus, it is necessary to *aggregate* information from multiple relation-time *temporal relationship* mentions to gain a complete temporal picture of a relation.

We focus on the aggregation of temporal information about relations within the context of the Temporal Slot-Filling (TSF) Task (Ji et al., 2011; Surdeanu, 2013). TSF focusses on a class of relations called *fluents* (Russell and Norvig, 2010), which are properties of named entities whose values may vary over time. Systems must succinctly describe all temporal information about each query relation R – e.g., title(Gbagbo, President) – available in a source document collection by assigning it a single, final temporal *four-tuple* (Amigo et al., 2011). Given a relation mention r of R and a time expression γ, a four-tuple $T_{\gamma}^{r} = \left\langle t^{(1)}, t^{(2)}, t^{(3)}, t^{(4)} \right\rangle$ characterizes their temporal relationship; namely, $t^{(1)}$ and $t^{(2)}$ represent the earliest and latest possible start date for R, while $t^{(3)}$ and $t^{(4)}$ represent the earliest and latest possible end dates, as inferred from the relation mention's context (sec. 3). For instance, a sentence indicating that Gbagbo was President on 2010-02-12 yields $\left\langle -\infty, \text{2010-02-12}, \text{2010-02-12}, +\infty \right\rangle$, while the sentence *"Gbagbo has been in power since 2000"* yields $\left\langle \text{2000-01-01}, \text{2000-12-31}, \text{2000-01-31}, +\infty \right\rangle$. The intuitively best aggregation of these four-tuples expresses what we learn from both texts, that the relation started in 2000 and remained ongoing at 2010-02-12, i.e. $\left\langle \text{2000-01-01}, \text{2010-02-12}, \text{2010-02-12}, +\infty \right\rangle$, with no clear indication as to its end. Straightforward cases like these were used to justify the simple aggregation methods used by all TSF systems to date (Surdeanu, 2013; Ji et al., 2011). However, in reality even humans often must deal with vague and/or conflicting temporal information across documents, and systems must furthermore deal with the fact that each of their temporal relationship classifications is potentially false.

To address the various properties of text and temporal representation that influence aggregation and affect final four-tuple quality, we first improve an existing gold standard dataset (sec. 4.1). We then describe two key factors affecting systems' aggregation performance: (1) erroneous classifications attributed high confidence by systems, and (2) a lack of relation-bounding classifications (sec. 4.2). We propose three methods to better prepare a relation's multiple mention context derived four-tuples for aggregation into a final four-tuple. The first applies simple rules to predicative nominal titles with explicit time information (e.g., *"former President"*), the second filters and re-labels four-tuples based on entity lifespan (sec. 5.3), and the third adds four-tuples based on mentions of relations other than, but temporally linked to, the query relation (sec. 5.4). We then discuss results and identify remaining challenges for aggregating temporal information across relation mentions (sec. 6 and 7). A Glossary of selected terms can be found in the appendix.

2 Related Work

The most similar work on temporal relation information aggregation are Wang et al. (2012), who use an Integer Linear Programming framework to enforce the validity of induced temporal relation information as well as enforce inter-relation constraints, and Dylla et al. (2013), who collect temporal information about relations, mostly about start and end times, using a temporal probabilistic data base framework to aggregate and enforce constraints based on relation argument existence. All TSF systems we are aware of have used either max-constrain or Validity-Ensured Incremental max-constrain aggregation algorithms (Surdeanu, 2013; Ji et al., 2011), which we cover in section 4. None we are aware of have applied background knowledge to constrain intermediate four-tuples (sec. 3) before or after aggregation. In this work we modified our previous work CUNYTSF (Artiles et al., 2011), which is the only publicly available TSF system we are aware of. CUNYTSF employs two supervised models, one based on a string kernel defined in terms of dependency paths between named entities involved in a relation and context time expressions, and the other based on bags-of-words derived from small windows surrounding these tokens and shallow dependency relations. CUNYTSF achieved the highest and second-highest scores of five systems in two TSF shared tasks (Surdeanu, 2013; Ji et al., 2011).

3 Temporal Slot Filling (TSF)

The 2013 Temporal Slot-Filling (TSF) (Surdeanu, 2013) task was part of the Knowledge Base Population (KBP) track of the Text Analysis Conference (TAC). Systems were given a list of 273 fluent relation instances as queries, each with a supporting document. Query relations were evenly distributed across relation types, which consisted of people's *titles, marriages, employments or memberships, and residences (city, state, and country)*, and companies' *top members or employees*. The task was to obtain a final four-tuple T_R for each query relation $R = \langle q,\ s \rangle$ using the source corpus for provenance. For each element in T_R a system must provide a document in which R is entailed, and offsets for the relation arguments (the query-entity q and slot-filler s) and the normalized time expression from which the four-tuple element is derived.

The KBP source collection consists of about 1 million newswire, 1 million web text, and 100,000 discussion forum documents. Gold standard annotation was obtained by annotators who, using a tool, searched the source corpus for documents that provide temporal information about each query relation. Given a mention r of R in a document d for which temporal information about R could be inferred, annotators assigned an intermediate temporal relationship label (Table 1) (Ji et al., 2011) to $\langle r, \gamma \rangle$, where γ is viewed as an interval of dates $[\gamma_s, \gamma_e]$ derived either based on (1) a normalized time expression in d, or (2) the document creation time of d. We denote the temporal extension of R at the day granularity $R_{ex} = [R_s, R_e]$, where R_s and R_e are the start and end dates of R. The intermediate label l mediates the relationship between γ and R_{ex}, characterizing a possible relationship between R and γ. [1] After systems submitted results for the shared task, any corresponding document not included in the original annotation

[1] We add AFTER_END* and BEFORE_START* but omit motivation due to space constraints.

that were determined to express R was exhaustively annotated for temporal information about R. A gold standard final four-tuple G_R is obtained for each R by applying an aggregation procedure (sec. 4.1) to the intermediate temporal relationship labels assigned to mention-time classification instances (Surdeanu, 2013).

In this work we adopt the evaluation metric used for the TSF shared task (Ji et al., 2011; Surdeanu, 2013).

Intermediate Relation	**four-tuple**
BEGINNING	$\langle \gamma_s, \gamma_e, \gamma_s, \infty \rangle$
ENDING	$\langle -\infty, \gamma_e, \gamma_s, \gamma_e \rangle$
BEG_AND_END	$\langle \gamma_s, \gamma_e, \gamma_s, \gamma_e \rangle$
WITHIN	$\langle -\infty, \gamma_e, \gamma_s, \infty \rangle$
THROUGHOUT	$\langle -\infty, \gamma_s, \gamma_e, \infty \rangle$
BEFORE_START	$\langle \gamma_e, \infty, \gamma_e, \infty \rangle$
AFTER_END	$\langle -\infty, \gamma_s, -\infty, \gamma_s \rangle$
BEFORE_START*	$\langle \gamma_s, \infty, \gamma_s, \infty \rangle$
AFTER_END*	$\langle -\infty, \gamma_e, -\infty, \gamma_e \rangle$
NONE	$\langle -\infty, \infty, -\infty, \infty \rangle$

Table 1: Intermediate temporal relationship function for $\langle r, \gamma \rangle$

Invalidity Source	Frequency
Conflicting Information	13
Multiple Instances	7
Wrong Intermediate Label	20
Vague Time Normalization	8
Other	8

Table 2: Reasons for Invalidity in Gold Standard Final Four-Tuples

4 Aggregating Intermediate Relations

Temporal information about instances of R must be aggregated to yield a complete temporal picture of the relation with respect to the background corpus. We denote with $I(R)$ the set of intermediate four-tuples associated with R. The purpose of the four-tuple representation is to be as accurate as possible in representing the extent to which a given corpus provides information about the start and end time of R, R_s and R_e, while preserving the vagueness inherent in the text. Each four-tuple element of $I(R)$ represents temporal information about R_s and/or R_e, most often with respect to the context associated with a particular mention r of R. Temporal information at a corpus level is derived via a process of aggregation over the elements of $I(R)$. In this section we describe how both human annotators and systems have approached aggregation.

4.1 Aggregating Manually Annotated Intermediate Relations

Gold standard four-tuples were obtained by applying the Max-Constrain (MC) algorithm (Equation 1) to each $I(R)$ obtained via manual annotation using the labels in Table 1 (Surdeanu, 2013; Ji et al., 2011).[2]

$$T_R = \langle max(t^{(1)}), min(t^{(2)}), max(t^{(3)}), min(t^{(4)}) \rangle \qquad (1)$$

Here, $max(t^{(k)})$ is the greatest $t^{(k)}$ from any intermediate four-tuple $T_r \in I(R)$, while $min(t^{(k)})$ is the least.

Let a four-tuple T be *valid* iff. $t^{(1)} \leq t^{(2)} \wedge t^{(3)} \leq t^{(4)} \wedge t^{(1)} \leq t^{(4)}$, and *correct* if $t^{(1)} \leq R_s \leq t^{(2)} \wedge t^{(3)} \leq R_e \leq t^{(4)}$. If R has only one start and one end date, and $R_s \leq R_e$, and each intermediate four-tuple $T_r^{\gamma} \in I(R)$ is valid and correct, then the final four-tuple obtained via MC is guaranteed to be valid and correct. Fifty-six gold standard final four-tuples were invalid and therefore discarded prior to evaluation (Surdeanu, 2013). We analyzed them by hand to determine the source of their invalidity (see Table 2). [3] We corrected instances until IMC (Algorithm 1) yielded a valid four-tuple.

[2] See http://surdeanu.info/kbp2013 for more details.

[3] Note that there may be more instances of each type described in table 2

[4] Here, $\max(t^{(i)} \leq x^{(i)}) := \max(\left\{t^{(i)} \in \mathbf{t}^{(i)} \middle| t^{(i)} \leq x^{(i)}\right\})$, where $\mathbf{t}^{(i)} := \left\{t^{(i)} \in T \middle| T \in I(R)\right\}$

Algorithm 1 Inclusive Max-Constrain (IMC)[4]

```
Require: $I(R) = \{T_0, T_1, \ldots, T_{N-1}\}$
Ensure: $T_R$
  $X \leftarrow$ max-constrain($I(R)$) $= \langle x^{(1)}, x^{(2)}, x^{(3)}, x^{(4)} \rangle$
  $Y \leftarrow \langle \max(t^{(1)} \leq x^{(2)}), \min(t^{(2)} \geq x^{(1)}), \max(t^{(3)} \leq x^{(4)}), \min(t^{(4)} \geq x^{(3)}) \rangle$
  $T_R \leftarrow \langle \max(t^{(1)} \leq y^{(2)}), \min(t^{(2)} \geq y^{(1)}), \max(t^{(3)} \leq y^{(4)}) \rangle, \min(t^{(4)} \geq y^{(3)})$
  return $T_R$
```

4.2 System Derived Intermediate Relations

As suggested in section 4.1, MC is sensitive to inconsistent four-tuples. In response to this all prior work that has not used MC to combine system-produced $I(R)$ has used an algorithm similar to Validity-Ensured Incremental (VEI) Max-Constrain (Algorithm 2) (Artiles et al., 2011). Here, $I(R)$ is ordered by classifier confidence and T_R is initialized as the trivial four-tuple and updated incrementally. Starting with the highest-confidence four-tuple $T_{R,0} \in I(R)$, MC is applied to $\{T_R, T_{R,i}\}$ to yield T^*. In a given iteration, T^* is only accepted as the updated T_R if it is valid. Intuitively, higher confidence intermediate four-tuples are more likely to be correct, thus the incremental algorithm tries to ensure that erroneous low-confidence four-tuples are less likely to be aggregated. In practice, however, a single high-confidence incorrect label can derail the entire process (sec. 5).

Algorithm 2 Validity-Ensured Incremental (VEI) Max-Constrain Aggregation to yield final four-tuple

```
Require: $I(R) = \{T_0, T_1, \ldots, T_{N-1}\}$
Ensure: $T_R = \langle t^{(1)}, t^{(2)}, t^{(3)}, t^{(4)} \rangle$
  $T_R \leftarrow \langle -\infty, \infty, -\infty, \infty \rangle$
  $i \leftarrow 0$
  while $i < N$ do
    $T^* \leftarrow \langle max(t^{(1)}, t_i^{(1)}), min(t^{(2)}, t_i^{(2)}), max(t^{(3)}, t_i^{(3)}), max(t^{(4)}, t_i^{(4)}) \rangle$ {Pairwise MC}
    if $t^{*(1)} \leq t^{*(2)} \wedge t^{*(3)} \leq t^{*(4)} \wedge t^{*(1)} \leq t^{*(4)}$ then
      $T_R \leftarrow T^*$ {Validity Check}
    end if
  end while
  return $T_R$
```

5 Challenges and Solutions

This section outlines our modifications to CUNYTSF, inspired by a preliminary error analysis. We implement three methods geared toward better preparing $I(R)$ for aggregation into a final four-tuple..

5.1 Preliminary Error Analysis

We ran the publicly available system CUNYTSF described in (Artiles et al., 2011) on the queries used in TSF2013, using the KBP2013 source collection, and evaluated against the corrected gold standard described in section 4.1. [5]

Error analysis revealed the main source of errors to be WITHIN labels with high confidence. To be exact, the final four-tuple for 116 queries (of 271) was influenced by a WITHIN label that yielded a $t^{(3)}$ later than the $g^{(4)}$ date, while 20 were influenced by WITHIN dates that were too early. Under VEI, once a labeled instance $\langle r, \gamma, \text{WITHIN} \rangle$ is aggregated into T_R, if $\gamma > R_e$ then any correctly labeled instance $\langle r, \gamma, \text{ENDING} \rangle$ will yield an invalid four-tuple and thus be rejected. (Similarly, correct BEGINNING labels will be blocked by incorrect WITHIN labels that are too early). Even correct WITHIN labels cannot set the corrupted aggregation back on track, since pairwise MC will always take the later $t^{(3)}$

[5] System downloaded from http://nlp.cs.rpi.edu/software.html

(algorithm 2). That said, WITHIN labels are often required to retrieve a complete temporal picture of a relation conveyed in a corpus. WITHIN is the most common intermediate label in the source collection, constituting 44% of correct labels, and furthermore, over half of the query relations require at least one WITHIN label to achieve the gold standard final four-tuple, with 10% relying solely on instances labeled WITHIN. To make matters worse, almost all TSF systems to date (except Garrido et al. (2013)) use neither the BEFORE_START* nor AFTER_END* labels in their intermediate temporal relationships classification models, even though high-confidence instances with those labels could prevent the sort of erroneous WITHIN labels alluded to above.

This analysis motivated three methods to curtail the extent to which aggregation-derailing four-tuples were included in $I(R)$ described in sections 5.2, 5.3, and 5.4. We favor VEI over IMC for system-derived $I(R)$ because IMC strongly relies on the assumption that there is a high probability of correctness for each intermediate relationship annotation.

5.2 Title Time of Predication

Nominal predicates are commonly used in English to refer to fluents. For example, attribution of a title to a person can be performed using a transitive verb or copula as in *"Serra was elected Governor"*, or *"Serra is the Governor"*, or as a Noun Phrase (NP) within a clause, as in *"Governor Jose Serra"* or *"Jose Serra, Governor, ..."* (among other ways). We refer to cases in which the subject and object of the relation are contained within a phrase headed by a Noun as Relational NP's (RNP).[6] For RNP that are mentions of fluent relations, there is a time of predication (TOP), i.e. a time at which the relation conveyed is asserted to hold, though this time is not overtly marked by tense or aspect (in English) as in the case of VP's. Tonhauser (2002)'s analysis assumes that the verbal time of predication (VTOP) is the "most salient" time in an utterance, thus relational NP's take their containing clause's verbal time of predication by default though contextual justification may override this tendency. We propose that in news the DCT is just as salient a time since the focus is centered on current affairs, an important entities are often "already introduced" into the discourse by virtue of being public figures. Ad-hoc analysis of the instances considered by CUNYTSF indicate that a compelling reason is required to override RNP's from taking both DCT and VTOP. For instance, in, *"O'Donnell ... suggested Wednesday that the Obama administration - particularly* ***Vice President Joe Biden****, who* ***represented*** *Delaware in the Senate for decades - was behind them"*, *"Vice President"* holds true at DCT, and rejects the VTOP of *"represented"*, presumably only based on logical inference: *no person is both Vice President and represents (a state) in the Senate at the same time*. Similarly, we know that the DCT (2010-08-04) is an invalid TOP in *"In November 2000, Chinese* ***President Jiang Zemin*** *paid a state visit to Laos, the first visit to Laos by a Chinese president"*, only because of world knowledge, or, *"The following is a chronology of major events in China- Laotian relations since 1990:"*, earlier in the document.

Though NP's lack tense and aspect, overt temporal modifiers such as *former, then-,* and *ex-* make explicit a *post-relational state* directly following an RNP's relation (Tonhauser, 2002).[7] The tendency for RNP's to take both the verbal predication time as well as the DCT extends to post-relational states. There are many examples in the corpus similar to the following: *"Former US President Bill Clinton and US journalists Euna Lee and Laura Ling returned Wednesday from North Korea, one day after North Korea's leader Kim Jong-Il pardoned the two women"*. Each RNP holds at the DCT, and *"Wednesday"*, as well as the day before that (the VTOP of *"pardoned"*). However, as for VTOP's further into the past, whether the post-relational state holds is less clear. For example, in, *"Secretary of State Hillary Rodham Clinton says former Philippines President Corazon Aquino "helped bring democracy back" to her country after years of authoritarian rule"*, we cannot rule out the possibility that Aquino helped bring democracy back *as President*; whether she did so *as former President* is left open, to be resolved by historical knowledge. This is likely because, unless the relation is of the "Grover Cleveland" type, once the relation becomes a "former" relation it will remain so thereafter.

[6] We adopt a Noun Phrase rather than a Determiner Phrase framework for simplicity.

[7] In this work we omit similar constructions that indicate a pre-relational state at the time of verbal predication, such as "future-", "soon-to-be", and "-elect". These words to not occur often in our data. That said, the extent to which their meanings are analogous to the overt temporal modifiers that introduce post-relational states is not clear, and requires further investigation.

The nature of the contexts that override default TOP for RNP's is complicated, and not well understood. In addition, determining VTOP automatically remains a difficult problem in and of itself (UzZaman et al., 2012). We have shown that newswire data contains relational NP's whose default times of predication - both DCT and verbal - are overridden by context. In addition, even post-relation states of modified RNP's may reject VTOP's. Post-relational states introduced by RNP's modified with *"former"*, *"then"*, and *"ex-"*, however, do appear to unambiguously take the DCT as a time of predication. Furthermore, we observe that CUNYTSF often incorrectly classifies modified RNP's introducing a post-relational state as expressing $\langle r, DCT, \textsc{within}\rangle$. To correct these errors we apply hand-written **Title Time of Predication Fix** rules to change the label for all such classification instances to AFTER_END* when the associated time expression is (or is closely related to) the DCT, and attribute 100% confidence to this new label. This correction both removes erroneous WITHIN labels and introduces labeled instances that bound query relations.

5.3 Entity Existence

VEI suffers when confidence values are inaccurate. For the relation spouse(Marylin Monroe, Arthur Miller), given the sentence, *"Editor Courtney Hodell said the book would include poems , photographs , reflections on third husband Arthur Miller and other men in Monroe 's life "*, a system is likely to mislabel $\langle r, \gamma\rangle$ as WITHIN, where γ is the document creation time 2010-04-27. The pattern *"husband s"* is a strong indicator of the WITHIN relationship for the spouse relation, so confidence for the resulting four-tuple $\langle -\infty, \text{2010-04-27}, \text{2010-04-27}, \infty\rangle$ is likely to be high. Once aggregated, it would be impossible to later aggregate $\langle -\infty, \text{1961-12-31}, \text{1961-01-01}, \text{1961-12-31}\rangle$ upon learning of the couple's divorce in 1961, since the proposed $T^* = \langle -\infty, \text{1961-12-31}, \text{2010-04-27}, \text{1961-12-31}\rangle$ is invalid. A basic clue that a WITHIN label should be changed to AFTER_END* is that q or s no longer exists (either the person has died or the business has dissolved).

To address this challenge we propose **Existence-based Correction and Filtering**. For each relation R we obtain the *existence four-tuple* E_R, by applying MC aggregation to the set of birth and death times in a knowledge base (KB) for the query-entity and slot-filler.[8] The KB is obtained via the Freebase API and scraping Wikipedia Infoboxes. We use a four-tuple instead of an interval of dates because birth and/or death information may not be available at the date granularity. Given the relation spouse(Jennifer Jones, Norton Simon) and the KB excerpt in Table 3, we obtain an existence constraint four-tuple $\langle \text{1919-03-02}, \text{1919-03-02}, \text{1993-06-02}, \text{1993-06-02}\rangle$.

Entity	Birth	Death
Jennifer Jones	1919-03-02	2009-12-17
Norton Simon	1907-02-05	1993-06-02

Table 3: Existence Information

We apply algorithm 3, where C contains classifier confidence for each labeled instance in $I(R)$. Above, $I(R)$ was introduced as a list of intermediate four-tuples for a relation R. In our approach, each of these four-tuples is derived deterministically (see Table 1). From here on (as in Algorithm 3) we allow a slightly abuse of notation in which $I(R)$ is viewed as a set of labeled classification instances, each of which yields a four-tuple for R. We omit pseudo-code to handle the analogous cases where instances are re-labeled BEFORE_START* based on the relative position of γ and ϵ_1.

5.4 Relation Precedence

The context of a relation mention often contains temporal information not explicitly tied to a time expression. For example, in, *"Myasnikovich will replace Sergei Sidorsky, who was prime minister since 2003"*, there is no date explicitly tied to the transition of power. Many titles are held by one person after another, in succession, without overlap. Intuitively, if we know the order in which several individuals held the same title then temporal information about one such relation can be used to constrain the other.

[8] For organization query-entities their foundation and defunct dates are considered their "birth" and "death" dates.

Algorithm 3 Existence Based Correction & Filtering Algorithm

```
Require: $I(R) = \{\langle\gamma_0, l_0\rangle, \ldots, \langle\gamma_k, l_k\rangle\}$; $C = \{c_0, \ldots, c_k\}$; $E_R = \langle\epsilon^{(1)}, \epsilon^{(2)}, \epsilon^{(3)}, \epsilon^{(4)}\rangle$
  while $i < N$ do
    if $\gamma_i.s \geq \epsilon_4 \wedge \neg(l_i = \text{NONE})$ then
      if $l_i = \text{ENDING} \wedge \gamma.s - \epsilon_4 \leq 31$ then
        $c_i \leftarrow 1.0$ {Most likely $R$ holds at the time of death}
      else
        $l_i \leftarrow \text{AFTER\_END*}$; $c_i \leftarrow 1.0$
      end if
    else if $\gamma_i.s \leq \epsilon_4 \leq \gamma_i.e \wedge \neg(l_i = \text{NONE})$ then
      if $l_i = \text{ENDING}$ then
        $c_i \leftarrow 1.0$ {Most likely $R$ holds at the time of death}
      else
        $l_i \leftarrow \text{AFTER\_END*}$; $c_i \leftarrow 1.0$
      end if
    end if
  end while
  return $I(R)$
```

To address this challenge we propose **Precedence-based Query Expansion and Re-labeling**. The title relation is well-represented in Wikipedia, and the infobox for many political title holders contains fields for *preceded by* and *succeeded by*, which specify the person that held the same title before and after the title holder in question. Given a title query R, we extracted the person who preceded and succeeded the query entity from the query entity's infobox (when available). Additional title relation *supporter* queries – R_{pre} and R_{suc}, respectively – were generated using these names, and the same title name as in the official query.[9]

After all classification instances are labeled and existence based correction is applied, we transform all labeled instances for supporter queries into labeled instances for official queries. Given a labeled instance $\langle r_x, \gamma, l\rangle$, where $x =$ pre or suc, we apply the mapping in Table 4 to yield the transformed labeled classification instance $\langle r, \gamma, l'\rangle$. Labeled supporter instances transformed into labeled official query instances are added to $I(R)$, the set of labeled instances for R. The set $I(R)$ is then passed to Aggregation (see Algorithm 2).

Supporter Label l	**Official label** l' ($x =$ pre)	**Official label** l' (when $x =$ suc)
NONE	NONE	NONE
BEFORE_START*	BEFORE_START*	NONE
AFTER_END*	NONE	AFTER_END*
All Others	BEFORE_START*	AFTER_END*

Table 4: Mapping to convert $\langle r_x, \gamma, l\rangle$ to $\langle r, \gamma, l'\rangle$, where x indicates whether the supporter query precedes or succeeds the official query

Just about any instance $\langle r_{\text{pre}}, \gamma, l\rangle$ yields $\langle r, \gamma, \text{BEFORE\_START*}\rangle$ because R_{pre} is known to both start and end before R starts. (And conversely $\langle r_{\text{suc}}, \gamma\rangle$ tends to yield AFTER_END* for $\langle r, \gamma\rangle$.) This is because the last (first) day of R_{pre} and all days before (after) it are guaranteed to be before (after) the start (end) of R. However, note that a AFTER_END* label for $\langle r_{\text{pre}}, \gamma\rangle$ yields NONE for R since dates after the end of R_{pre} may be before, during, or after R. For example, the headline, *"Former President*

[9] In general, knowing that two relations stand in a particular interval relation to one another allows us to posit constraints on one relation upon discovering temporal information about the other. We apply this intuition to the title relation in this work since the information is readily available in a structured form (i.e., the *preceded by* and *succeeded by* fields in Wikipedia info boxes).

Lee Teng-hui on visit in Japan Tokyo", while clearly indicating AFTER_END$*$ for R_{pre} tells us very little about the relationship between the document creation time and R.

6 Results and Analysis

We scored the output for five conditions using the modified gold standard (section 4.1). TF means that title time of predication fix was applied (section 5.2), EC means existence corrections were applied, and Pr means that precedence-based query expansion was applied (section 5.4).

System	P	R	F
CUNYTSF	.337	.294	.314
CUNYTSF + TF	.341	.298	.318
CUNYTSF + EC	.349	.305	.326
CUNYTSF + TF + EC	.353	.309	.329
CUNYTSF + TF + EC + Pr	**.360**	**.315**	**.336**

Table 5: Results calculated using official TSF2013 scorer against corrected gold standard (sec. 4.1), with `anydoc` and `ignore-offsets` parameters set to true, augmented to calculate recall and precision

6.1 Title Time of Predication Fix

The gold standard for `title` had 142 non-infinity tuple element outputs of the form $\langle R, i, t^{(i)} \rangle$. The baseline output had 80 values while baseline + TF had 91. Applying TF, 10 baseline outputs were replaced while 11 were added. In most cases erroneous WITHIN labels are corrected by inserting high-confidence AFTER_END$*$ into $I(R)$. In some cases this allows a correct $t^{(3)}$ to replace a later, incorrect $t^{(3)}$ that came from an erroneous WITHIN label. It is important to note that while some changes barely affect F-measure, they are important because they allow for correct information that would have otherwise been blocked to be aggregated. For example, a bad baseline WITHIN for *"General Prosecutor 's Office of Kyrgyzstan on Tuesday charged the country's former Prime Minister Igor Chudinov with abuse of power"* had blocked a correct WITHIN for *"Kyrgyz Prime Minister Igor Chudinov left Beijing Thursday evening"* - removing this block allowed $t^{(3)}$ to change from 2010-05-04 to 2009-10-14, which is the gold standard value.

6.2 Existence-based Correction and Filtering

Most changes made from existence constraints are beneficial both in terms of an increase in F-measure and in blocking the aggregation of incorrect information. For instance, it is difficult to prevent labeling the following sentence with WITHIN for DCT: *"The **London** home of composer George Frideric Handel is holding an exhibition about its other famous resident – **Jimi Hendrix**"*, but the document context permits AFTER_END$*$, given *"**Hendrix** died in **London** on Sept. 18 , 1970"*. Given the existence constraint we label the instance AFTER_END$*$.

On the other hand, in some cases we erroneously change WITHIN to BEFORE_START$*$ using existence constraints, but this type of change does little damage. For example, the fact that CNN was founded on 1980-06-01 changes the label on 1980 from WITHIN to BEFORE_START$*$ for EMPLOYEE(Novak, CNN), given *"Novak , editor of the Evans-Novak Political Report , is perhaps best known as a co-host of several of CNN 's political talk shows , where he often jousted with liberal guests from 1980 to 2005"*. We set $t^{(1)} = 1980\text{-}01\text{-}01$ which does not block later inclusion of a correct $\langle R, 1980, \text{BEGINNING} \rangle$, which would set $t^{(1)} = 1980\text{-}01\text{-}01$ if it were not already set, and does set $t^{(2)} = 1980\text{-}12\text{-}31$. Changing this relation's label from WITHIN to START is not a catastrophic error because it allows for a finer grained, correct start date to be aggregated using VEI (see Algorithm 2) to yield a superior final four-tuple (though CUNYTSF finds no suitable candidates to facilitate this).

6.3 Precedence-based Query Expansion & Re-labeling

Output for affected official queries were improved simply because supporter queries were accurately labeled. For example, *"Kim Choongsoo, Korea's Central Bank Governor, said here on Thursday his nation's economic situation was getting better"* provides a $t^{(4)}$ value for `title`(Lee Seong-tae, Governor) due given the successor relation.

Some gains from label transformation are only possible given the title time of predication fix. For example, multiple instances of *"former president Chen Shui-bian"* and *"Former President Lee Teng-hui"* were converted from WITHIN to AFTER_END* for their respective relations. Because Chen succeeded Lee, the latter instances were transformed to NONE instances for `title`(Chen, President) using Table 4. [10] Changing these labels to NONE made room for a valid $t^{(3)} =$ 2000-01-01 based converting the WITHIN for `title`(Lee, President) to BEFORE_START* for `title`(Chen, President) given, *"... since former President Lee Teng-hui promulgated it 19 years ago, Wang said, and the [DPP] did not try to make any changes to the framework during its eight-year rule between 2000 and 2008 either"*.

Label transformation is robust to misclassification. For example, any of BEFORE_START*, BEGINNING, WITHIN, or ENDING for a predecessor relation R_{pre} will map to before_start* for R. But other types of errors propagate and can lead to disastrous results. For example, due to a normalization quirk *"Utatu President George Strauss"* is recognized as *"Johannes Rau"*, thus the relation `title`(Rau, President) was assigned WITHIN at DCT, which is converted to a BEFORE_START* for Horst Kohler, Rau's successor.

A deeper problem that can lead to error propagation is that fact one person can have the same title in different contexts. When a title is attributed to a person there is often a geo-political or organization entity involved. Mentions that fail to include this third entity are ambiguous; often, this information needs to be inferred from other context sentences. Such errors may be propagated from supporter to official queries. For example, *"Francophonie president Abdou Diouf of Senegal ... "* appears to support the `title`(Abdou Diouf, President). Diouf preceded Abdoulaye Wade as President of Senegal, but the context in question (inaccurately) refers to Diouf's leadership position of Secretary-General (not President) of Organisation internationale de la Francophonie, thus an erroneous BEFORE_START* is aggregated, blocking a correctly labeled (less confident) $\langle r, 2000, \text{START}\rangle$.

7 Conclusion

We have analyzed within the particular context of TSF the process of aggregating partially-specified temporal information about relations across documents. Our analysis and and results indicate that text mentions of relations often ground only a portion of the referent relation in time and that correct interpretation relies on background knowledge about relation participants. In future work we plan a more rigorous data-driven study of nominal time of predication and to attack more ambiguous context-sensitive cases. In addition we aim to induce relation order from text automatically to multiple relation types as well as events.

Acknowledgments

This work was supported by the U.S. Army Research Laboratory under Cooperative Agreement No. W911NF-09-2-0053 (NS-CTA), and in addition the U.K. Ministry of Defense under Agreement No. W911NF-06-3-0001 (ITA), U.S. NSF CAREER Award under Grant IIS-0953149, U.S. DARPA Award No. FA8750-13-2-0041 in the Deep Exploration and Filtering of Text (DEFT) Program, IBM Faculty Award, Google Research Award and RPI faculty start-up grant. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

[10] Had the title fix not been applied these WITHIN labels would have been converted to BEFORE_START*.

References

Enrique Amigo, Artiles Javier, Qi Li, and Heng Ji. 2011. An evaluation framework for aggregated temporal information extraction. In *Pric SIGIR2011 Workshop on Entity-Oriented Search.*

Javier Artiles, Qi Li, Taylor Cassidy, and Heng Ji. 2011. Temporal slot filling system description. In *Proc. Text Analytics Conference (TAC2011).*

Maximilian Dylla, Iris Miliaraki, and Martin Theobald. 2013. A temporal-probabilistic database model for information extraction. *Proceedings of the VLDB Endowment*, 6(14):1810–1821.

Guillermo Garrido, Anselmo Penas, and Bernardo Cabaleiro. 2013. Uned slot filling and temporal slot filling systems at tac kbp 2013. system description. In *Proc. Text Analytics Conference (TAC2013).*

Heng Ji, Ralph Grishman, and Hoa Trang Dang. 2011. An overview of the tac2011 knowledge base population track. In *Proc. Text Analytics Conference (TAC2011).*

Stuart J. Russell and Peter Norvig. 2010. *Artificial Intelligence - A Modern Approach (3. internat. ed.).* Pearson Education.

Mihai Surdeanu. 2013. An overview of the tac2013 knowledge base population track. In *Proc. Text Analytics Conference (TAC2013).*

Judith Tonhauser. 2002. A dynamic semantic account of the temporal interpretation of noun phrases. In *Proceedings of SALT*, volume 12, pages 286–305.

Naushad UzZaman, Hector Llorens, James F. Allen, Leon Derczynski, Marc Verhagen, and James Pustejovsky. 2012. Tempeval-3: Evaluating events, time expressions, and temporal relations. *CoRR*, abs/1206.5333.

Yafang Wang, Maximilian Dylla, Marc Spaniol, and Gerhard Weikum. 2012. Coupling label propagation and constraints for temporal fact extraction. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 233–237. Association for Computational Linguistics.

Appendix A. Glossary of Selected Terms

Fluent Relation: A property of a person or organization whose value may change over time. For example, a person's employer.

Temporal Extension: For a relation R, the temporal extension is the interval $[R_s, R_e]$, which represents the period of time between and including the start date R_s and end date R_e of the relation.

Relation Mention: An excerpt of text that expresses a relation.

Time Expression: An excerpt of text that refers to a portion of time, such as "Tuesday" or "next year".

Normalized Time Expression: The portion of time indicated by a time expression expressed in a standard form.

Granularity: The level at which a portion of time is expressed, in terms of calendar and clock units. For example, years are of a coarser granularity than days.

Temporal Four-tuple: For a relation R, a temporal four-tuple $T_R = \left\langle t^{(1)}, t^{(2)}, t^{(3)}, t^{(4)} \right\rangle$ represents an assertion that, based on some evidence, the start date for R is between $t^{(1)}$ and $t^{(2)}$, and its end date is between $t^{(3)}$ and $t^{(4)}$.

Final Temporal Four-tuple: The four-tuple assigned to R (by an annotator or system) after aggregating all temporal information about R.

Valid Temporal Four-tuple: A four-tuple $T = \left\langle t^{(1)}, t^{(2)}, t^{(3)}, t^{(4)} \right\rangle$ is valid if and only if iff. $t^{(1)} \leq t^{(2)} \wedge t^{(3)} \leq t^{(4)} \wedge t^{(1)} \leq t^{(4)}$.

Correct Temporal Four-tuple: A temporal four-tuple $T_R = \left\langle t^{(1)}, t^{(2)}, t^{(3)}, t^{(4)} \right\rangle$ if and only if $t^{(1)} \leq R_s \leq t^{(2)} \wedge t^{(3)} \leq R_e \leq t^{(4)}$

Intermediate Temporal Relationship: Given a relation mention r of relation R and a normalized time expression γ (viewed as a temporal interval), the intermediate temporal relationship between the two characterizes the relationships between the end points of γ and the endpoints of the temporal extension of R, namely γ_s, γ_e, R_s, and R_e. In this work, each intermediate temporal relationship used serves as a mapping from temporal interval to four-tuple (see Table 1 for the relationships used in this work and their mappings).

Intermediate Temporal Four-tuple Set: For a relation R, a system or annotator may derive an intermediate temporal four-tuple for each relation mention r and a corresponding time expression γ by based on an intermediate temporal relationship expressed between the two. The elements of each intermediate four-tuple are derived using the mapping in Table 1. We denote the set of intermediate temporal four-tuples for R as $I(R)$.

Query Relation: A relation that serves as input to a TSF system tasked with returning a final temporal four-tuple for that relation.

Relational Noun Phrase: A noun phrase that expresses a relation. For example, "President Obama" expresses a relation that "Obama"'s title is "President".

Time of Predication: For a given predicate, the time of predication is a time interval for which the predicate is asserted to apply to a specified set of arguments.

Post-relational State: A state immediately following the end of a relation characterized by the relation now longer holding. For example, prepending a title with "former", as in "former President X", introduces a state characterized by X no longer holding the title President.

Temporally Linked Relations: Two relations are temporally linked if their temporal extensions are not independent. For example, if it is known that one's end precedes the other's start.

Provenance: The relevant text that supports the output.