Coupling Label Propagation and Constraints for Temporal Fact Extraction

Yafang Wang, Maximilian Dylla, Marc Spaniol and Gerhard Weikum
Max Planck Institute for Informatics, Saarbrücken, Germany
{ywang|mdylla|mspaniol|weikum}@mpi-inf.mpg.de

Abstract

The Web and digitized text sources contain a wealth of information about named entities such as politicians, actors, companies, or cultural landmarks. Extracting this information has enabled the automated construction of large knowledge bases, containing hundred millions of binary relationships or attribute values about these named entities. However, in reality most knowledge is transient, i.e. changes over time, requiring a temporal dimension in fact extraction. In this paper we develop a methodology that combines label propagation with constraint reasoning for temporal fact extraction. Label propagation aggressively gathers fact candidates, and an Integer Linear Program is used to clean out false hypotheses that violate temporal constraints. Our method is able to improve on recall while keeping up with precision, which we demonstrate by experiments with biography-style Wikipedia pages and a large corpus of news articles.

1 Introduction

In recent years, automated fact extraction from Web contents has seen significant progress with the emergence of freely available knowledge bases, such as DBpedia (Auer et al., 2007), YAGO (Suchanek et al., 2007), TextRunner (Etzioni et al., 2008), or ReadTheWeb (Carlson et al., 2010a). These knowledge bases are constantly growing and contain currently (by example of DBpedia) several million entities and half a billion facts about them. This wealth of data allows to satisfy the information needs of advanced Internet users by raising queries from keywords to entities. This enables queries like "Who is married to Prince Charles?" or "Who are the teammates of Lionel Messi at FC Barcelona?".

However, factual knowledge is highly ephemeral: Royals get married and divorced, politicians hold positions only for a limited time and soccer players transfer from one club to another. Consequently, knowledge bases should be able to support more sophisticated *temporal* queries at *entity-level*, such as "Who have been the spouses of Prince Charles before 2000?" or "Who are the teammates of Lionel Messi at FC Barcelona in the season 2011/2012?". In order to achieve this goal, the next big step is to distill *temporal knowledge* from the Web.

Extracting temporal facts is a complex and time-consuming endeavor. There are "conservative" strategies that aim at high precision, but they tend to suffer from low recall. On the contrary, there are "aggressive" approaches that target at high recall, but frequently suffer from low precision. To this end, we introduce a method that allows us to gain maximum benefit from both "worlds" by "aggressively" gathering fact candidates and subsequently "cleaning-up" the incorrect ones. The salient properties of our approach and the novel contributions of this paper are the following:

- A temporal fact extraction strategy that is able to efficiently gather thousands of fact candidates based on a handful of seed facts.
- An ILP solver incorporating constraints on temporal relations among events (e.g., marriage of a person must be non-overlapping in time).
- Experiments on real world news and Wikipedia articles showing that we gain recall while keeping up with precision.

2 Related Work

Recently, there have been several approaches that aim at the extraction of temporal facts for the automated construction of large knowledge bases, but

Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pages 233–237,
Jeju, Republic of Korea, 8-14 July 2012.

time-aware fact extraction is still in its infancy. An approach toward fact extraction based on coupled semi-supervised learning for information extraction (IE) is NELL (Carlson et al., 2010b). However, it does neither incorporate constraints nor temporality. TIE (Ling and Weld, 2010) binds time-points of events described in sentences, but does not disambiguate entities or combine observations to facts. A pattern-based approach for temporal fact extraction is PRAVDA (Wang et al., 2011), which utilizes label propagation as a semi-supervised learning strategy, but does not incorporate constraints. Similarly, TOB is an approach of extracting temporal business-related facts from free text, which requires deep parsing and does not apply constraints as well (Zhang et al., 2008). In contrast, CoTS (Talukdar et al., 2012) introduces a constraint-based approach of coupled semi-supervised learning for IE, however not focusing on the extraction part. Building on TimeML (Pustejovsky et al., 2003) several works (Verhagen et al., 2005; Mani et al., 2006; Chambers and Jurafsky, 2008; Verhagen et al., 2009; Yoshikawa et al., 2009) identify temporal relationships in free text, but don't focus on fact extraction.

3 Framework

Facts and Observations. We aim to extract factual knowledge transient over time from free text. More specifically, we assume *time* $\mathcal{T} = [0, T_{max}]$ to be a finite sequence of time-points with yearly granularity. Furthermore, a *fact* consists of a relation with two typed arguments and a time-interval defining its validity. For instance, we write $worksForClub(Beckham, RMadrid)@[2003, 2008)$ to express that Beckham played for Real Madrid from 2003 to 2007. Since sentences containing a fact and its full time-interval are sparse, we consider three kinds of textual observations for each relation, namely *begin*, *during*, and *end*. "Beckham signed for Real Madrid from Manchester United in 2003." includes both the *begin* observation of Beckham being with Real Madrid as well as the *end* observation of working for Manchester. A *positive seed fact* is a valid fact of a relation, while a *negative seed fact* is incorrect (e.g., for relation $worksForClub$, a *positive seed fact* is $worksForClub(Beckham, RMadrid)$, while $worksForClub(Beckham, BMunich)$ is a *negative seed fact*).

Framework. As depicted in Figure 1, our framework is composed of four stages, where the first collects candidate sentences, the second mines patterns from the candidates sentences, the third extracts temporal facts from the sentences utilizing the patterns and the last removes noisy facts by enforcing constraints.

Preprocessing. We retrieve all sentences from the corpus comprising at least two entities and a temporal expression, where we use YAGO for entity recognition and disambiguation (cf. (Hoffart et al., 2011)).

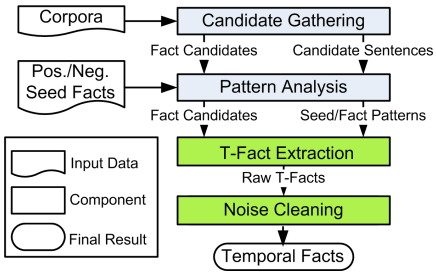

Figure 1: System Overview

Pattern Analysis. A *pattern* is a n-gram based feature vector. It is generated by replacing entities by their types, keeping only stemmed nouns, verbs converted to present tense and the last preposition. For example, considering "Beckham signed for Real Madrid from Manchester United in 2003." the corresponding pattern for the *end* occurrence is "sign for CLUB from". We quantify the *strength* of each pattern by investigating how frequent the pattern occurs with seed facts of a particular relation and how infrequent it appears with negative seed facts.

Fact Candidate Gathering. Entity pairs that co-occur with patterns whose strength is above a minimum threshold become fact candidates and are fed into the next stage of label propagation.

4 T-Fact Extraction

Building on (Wang et al., 2011) we utilize Label Propagation (Talukdar and Crammer, 2009) to determine the relation and observation type expressed by each pattern.

Graph. We create a graph $G = (\mathcal{V}_F \dot{\cup} \mathcal{V}_P, \mathcal{E})$ having one vertex $v \in \mathcal{V}_F$ for each fact candidate observed in the text and one vertex $v \in \mathcal{V}_P$ for each pattern. Edges between $\mathcal{V}_F$ and $\mathcal{V}_P$ are introduced whenever a fact candidate appeared with a pattern. Their weight is derived from the co-occurrence frequency. Edges

among $\mathcal{V}_P$ nodes have weights derived from the n-gram overlap of the patterns.

Labels. Moreover, we use one label for each observation type (*begin*, *during*, and *end*) of each relation and a dummy label representing the unknown relation.

Objective Function. Let $\mathbf{Y} \in \mathbb{R}_+^{|\mathcal{V}|\times|Labels|}$ denote the graph's initial label assignment, and $\widehat{\mathbf{Y}} \in \mathbb{R}_+^{|\mathcal{V}|\times|Labels|}$ stand for the estimated labels of all vertices, $\mathbf{S}_l$ encode the seed's weights on its diagonal, and $\mathbf{R}_{*l}$ contain zeroes except for the dummy label's column. Then, the objective function is:

$$\mathcal{L}(\widehat{\mathbf{Y}}) = \sum_{\ell} \begin{bmatrix} (\mathbf{Y}_{*\ell} - \widehat{\mathbf{Y}}_{*\ell})^T \mathbf{S}_\ell (\mathbf{Y}_{*\ell} - \widehat{\mathbf{Y}}_{*\ell}) \\ +\mu_1 \widehat{\mathbf{Y}}_{*\ell}^T \mathbf{L} \widehat{\mathbf{Y}}_{*\ell} + \mu_2 \|\widehat{\mathbf{Y}}_{*\ell} - \mathbf{R}_{*\ell}\|^2 \end{bmatrix} \quad (1)$$

Here, the first term $(\mathbf{Y}_{*\ell} - \widehat{\mathbf{Y}}_{*\ell})^T \mathbf{S}_\ell (\mathbf{Y}_{*\ell} - \widehat{\mathbf{Y}}_{*\ell})$ ensures that the estimated labels approximate the initial labels. The labeling of neighboring vertices is smoothed by $\mu_1 \widehat{\mathbf{Y}}_{*\ell}^T \mathbf{L} \widehat{\mathbf{Y}}_{*\ell}$, where $\mathbf{L}$ refers to the Laplacian matrix. The last term is a L2 regularizer.

5 Cleaning of Fact Candidates

To prune noisy t-facts, we compute a consistent subset of t-facts with respect to temporal constraints (e.g. joining a sports club takes place before leaving a sports club) by an Integer Linear Program (ILP).

Variables. We introduce a variable $x_r \in \{0, 1\}$ for each t-fact candidate $r \in \mathcal{R}$, where 1 means the candidate is valid. Two variables $x_{f,b}, x_{f,e} \in [0, T_{max}]$ denote begin (b) and end (e) of time-interval of a fact $f \in \mathcal{F}$. Note, that many t-fact candidates refer to the same fact f, since they share their entity pairs.

Objective Function. The objective function intends to maximize the number of valid raw t-facts, where w_r is a weight obtained from the previous stage:

$$\max \sum_{r \in \mathcal{R}} w_r \cdot x_r$$

Intra-Fact Constraints. $x_{f,b}$ and $x_{f,e}$ encode a proper time-interval by adding the constraint:

$$\forall f \in \mathcal{F} \quad x_{f,b} < x_{f,e}$$

Considering only a single relation, we assume the sets $\mathcal{R}_b$, $\mathcal{R}_d$, and $\mathcal{R}_e$ to comprise its t-fact candidates with respect to the *begin*, *during*, and *end* observations. Then, we introduce the constraints

$$\forall l \in \{b, e\}, r \in \mathcal{R}_l \quad t_l \cdot x_r \leq x_{f,l} \quad (2)$$

$$\forall l \in \{b, e\}, r \in \mathcal{R}_l \quad x_{f,l} \leq t_l \cdot x_r + (1 - x_r) T_{max} \quad (3)$$

$$\forall r \in \mathcal{R}_d \quad x_{f,b} \leq t_b \cdot x_r + (1 - x_r) T_{max} \quad (4)$$

$$\forall r \in \mathcal{R}_d \quad t_e \cdot x_r \leq x_{f,e} \quad (5)$$

where f has the same entity pair as r and t_b, t_e are begin and end of r's time-interval. Whenever x_r is set to 1 for *begin* or *end* t-fact candidates, Eq. (2) and Eq. (3) set the value of $x_{f,b}$ or $x_{f,e}$ to t_b or t_e, respectively. For each *during* t-fact candidate with $x_r = 1$, Eq. (4) and Eq. (5) enforce $x_{f,b} \leq t_b$ and $t_e \leq x_{f,e}$.

Inter-Fact Constraints. Since we can refer to a fact f's time interval by $x_{f,b}$ and $x_{f,e}$ and the connectives of Boolean Logic can be encoded in ILPs (Karp, 1972), we can use all temporal constraints expressible by Allen's Interval Algebra (Allen, 1983) to specify inter-fact constraints. For example, we leverage this by prohibiting marriages of a single person from overlapping in time.

Previous Work. In comparison to (Talukdar et al., 2012), our ILP encoding is time-scale invariant. That is, for the same data, if the granularity of $\mathcal{T}$ is changed from months to seconds, for example, the size of the ILP is not affected. Furthermore, because we allow all relations of Allen's Interval Algebra, we support a richer class of temporal constraints.

6 Experiments

Corpus. Experiments are conducted in the soccer and the celebrity domain by considering the *worksForClub* and *isMarriedTo* relation, respectively. For each person in the "FIFA 100 list" and "Forbes 100 list" we retrieve their Wikipedia article. In addition, we obtained about 80,000 documents for the soccer domain and 370,000 documents for the celebrity domain from BBC, The Telegraph, Times Online and ESPN by querying Google's News Archive Search[1] in the time window from 1990-2011. All hyperparameters are tuned on a separate data-set.

Seeds. For each relation we manually select the 10 positive and negative fact candidates with highest occurrence frequencies in the corpus as seeds.

Evaluation. We evaluate *precision* by randomly sampling 50 (*isMarriedTo*) and 100 (*worksForClub*) facts for each observation type and manually evaluating them against the text documents. All experimental data is available for download from our website[2].

6.1 Pipeline vs. Joint Model

Setting. In this experiment we compare the performance of the pipeline being stages 3 and 4 in Figure

[1] news.google.com/archivesearch

[2] www.mpi-inf.mpg.de/yago-naga/pravda/

1 and a joint model in form of an ILP solving the t-fact extraction and noise cleaning at the same time. Hence, the joint model resembles (Roth and Yih, 2004) extended by Section 5's temporal constraints.

Relation	Observation	Label Propagation		ILP for T-Fact Extraction		
		Precision	# Obs.	Precision	# Obs.	
worksForClub	begin	80%	2537	81%	2426	Without Noise Cleaning
	during	78%	2826	86%	1153	
	end	65%	440	50%	550	
isMarriedTo	begin	52%	195	28%	232	
	during	76%	92	6%	466	
	end	62%	50	2%	551	
worksForClub	begin	85%	2469	87%	2076	With Noise Cleaning
	during	85%	2761	79%	1434	
	end	74%	403	72%	275	
isMarriedTo	begin	64%	177	74%	67	
	during	79%	89	88%	61	
	end	70%	47	71%	28	

Table 1: Pipeline vs. Joint Model

Results. Table 1 shows the results on the pipeline model (lower-left), joint model (lower-right), label-propagation w/o noise cleaning (upper-left), and ILP for t-fact extraction w/o noise cleaning (upper-right).
Analysis. Regarding the upper part of Table 1 the pattern-based extraction works very well for *worksForClub*, however it fails on *isMarriedTo*. The reason is, that the types of *worksForClub* distinguish the patterns well from other relations. In contrast, *isMarriedTo*'s patterns interfere with other person-person relations making constraints a decisive asset. When comparing the joint model and the pipeline model, the former sacrifices recall in order to keep up with the latter's precision level. That is because the joint model's ILP decides with binary variables on which patterns to accept. In contrast, label propagation addresses the inherent uncertainty by providing label assignments with confidence numbers.

6.2 Increasing Recall

Setting. In a second experiment, we move the t-fact extraction stage away from high precision towards higher recall, where the successive noise cleaning stage attempts to restore the precision level.
Results. The columns of Table 2 show results for different values of μ_1 of Eq. (1). From left to right, we used $\mu_1 = e^{-1}, 0.6, 0.8$ for *worksForClub* and $\mu_1 = e^{-2}, e^{-1}, 0.6$ for *isMarriedTo*. The table's upper part reports on the output of stage 3, whereas the lower part covers the facts returned by noise cleaning.
Analysis. For the conservative setting label propagation produces high precision facts with only few inconsistencies, so the noise cleaning stage has no effect, i.e. no pruning takes place. This is the setting usual pattern-based approaches without cleaning stage are working in. In contrast, for the standard setting (coinciding with Table 1's left column) stage 3 yields less precision, but higher recall. Since there are more inconsistencies in this setup, the noise cleaning stage accomplishes precision gains compensating for the losses in the previous stage. In the relaxed setting precision drops too low, so the noise cleaning stage is unable to figure out the truly correct facts. In general, the effects on *worksForClub* are weaker, since in this relation the constraints are less influential.

		Conservative		Standard		Relaxed		
		Prec.	# Obs.	Prec.	# Obs.	Prec.	# Obs.	
worksForClub	begin	83%	2443	80%	2537	80%	2608	Without Noise Cleaning
	during	81%	2523	78%	2826	76%	2928	
	end	77%	377	65%	440	62%	501	
isMarriedTo	begin	72%	112	52%	195	44%	269	
	during	90%	63	76%	92	52%	187	
	end	67%	37	62%	50	36%	116	
worksForClub	begin	83%	2389	85%	2469	84%	2536	With Noise Cleaning
	during	88%	2474	85%	2761	75%	2861	
	end	79%	349	72%	403	70%	463	
isMarriedTo	begin	72%	111	64%	177	46%	239	
	during	90%	62	79%	89	54%	177	
	end	69%	36	68%	47	38%	110	

Table 2: Increasing Recall.

7 Conclusion

In this paper we have developed a method that combines label propagation with constraint reasoning for temporal fact extraction. Our experiments have shown that best results can be achieved by applying "aggressive" label propagation with a subsequent ILP for "clean-up". By coupling both approaches we achieve both high(er) precision and high(er) recall. Thus, our method efficiently extracts high quality temporal facts at large scale.

Acknowledgements

This work is supported by the 7^{th} Framework IST programme of the European Union through the focused research project (STREP) on Longitudinal Analytics of Web Archive data (LAWA) under contract no. 258105.

References

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

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *In 6th Intl Semantic Web Conference, Busan, Korea*, pages 11–15. Springer.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. 2010a. Toward an architecture for never-ending language learning. In *AAAI*, pages 1306–1313.

Andrew Carlson, Justin Betteridge, Richard C. Wang, Estevam R. Hruschka Jr., and Tom M. Mitchell. 2010b. Coupled semi-supervised learning for information extraction. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining (WSDM 2010)*.

Nathanael Chambers and Daniel Jurafsky. 2008. Jointly combining implicit constraints improves temporal ordering. In *EMNLP*, pages 698–706.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. 2008. Open information extraction from the web. *Commun. ACM*, 51(12):68–74, December.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In *Proc. of EMNLP 2011: Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK, July 2731*, pages 782–792.

Richard M. Karp. 1972. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103.

Xiao Ling and Daniel S. Weld. 2010. Temporal information extraction. In *Proceedings of the AAAI 2010 Conference*, pages 1385 – 1390, Atlanta, Georgia, USA, July 11-15. Association for the Advancement of Artificial Intelligence.

Inderjeet Mani, Marc Verhagen, Ben Wellner, Chong Min Lee, and James Pustejovsky. 2006. Machine learning of temporal relations. In *In ACL-06*, pages 17–18.

James Pustejovsky, José M. Castaño, Robert Ingria, Roser Sauri, Robert J. Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R. Radev. 2003. TimeML: Robust specification of event and temporal expressions in text. In *New Directions in Question Answering*, pages 28–34.

Dan Roth and Wen-Tau Yih, 2004. *A Linear Programming Formulation for Global Inference in Natural Language Tasks*, pages 1–8.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *WWW '07: Proceedings of the 16th International Conference on World Wide Web*, pages 697–706, New York, NY, USA. ACM.

Partha Pratim Talukdar and Koby Crammer. 2009. New regularized algorithms for transductive learning. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part II*, ECML PKDD '09, pages 442–457, Berlin, Heidelberg. Springer-Verlag.

Partha Pratim Talukdar, Derry Wijaya, and Tom Mitchell. 2012. Coupled temporal scoping of relational facts. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining (WSDM)*, Seattle, Washington, USA, February. Association for Computing Machinery.

Marc Verhagen, Inderjeet Mani, Roser Sauri, Robert Knippen, Seok Bae Jang, Jessica Littman, Anna Rumshisky, John Phillips, and James Pustejovsky. 2005. Automating temporal annotation with TARSQI. In *ACL '05: Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, pages 81–84, Morristown, NJ, USA. Association for Computational Linguistics.

Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Jessica Moszkowicz, and James Pustejovsky. 2009. The tempeval challenge: identifying temporal relations in text. *Language Resources and Evaluation*, 43:161–179.

Yafang Wang, Bin Yang, Lizhen Qu, Marc Spaniol, and Gerhard Weikum. 2011. Harvesting facts from textual web sources by constrained label propagation. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 837–846, New York, NY, USA. ACM.

Katsumasa Yoshikawa, Sebastian Riedel, Masayuki Asahara, and Yuji Matsumoto. 2009. Jointly identifying temporal relations with markov logic. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, pages 405–413, Stroudsburg, PA, USA. Association for Computational Linguistics.

Qi Zhang, Fabian Suchanek, and Gerhard Weikum. 2008. TOB: Timely ontologies for business relations. In *11th International Workshop on Web and Databases 2008 (WebDB 2008)*, Vancouver, Canada. ACM.