Machine Learning Approaches for Temporal Information Extraction: A Comparative Study

Oleksandr Kolomiyets, Marie-Francine Moens

Department of Computer Science
Katholieke Universiteit Leuven
Celestijnenlaan 200A
3001 Heverlee, Belgium
oleksandr.kolomiyets@cs.kuleuven.be
sien.moens@cs.kuleuven.be

Abstract: Temporal expressions are important structures in natural language. In order to understand text, temporal expressions have to be extracted and normalized to ISO-based values. For these purposes rule-based and machine learning techniques were proposed. In this paper we present and compare two approaches for automatic recognition of temporal expressions in free text, based on a supervised machine learning approach and trained on an annotated corpus for temporal information, namely TimeBank. The first approach performs a token-by-token classification following B-I-O encoding. The second one does a binary constituent-based classification of chunk phrases. Our experiments demonstrate that on the TimeBank corpus the constituent-based classification performs better than the token-based one. It achieves F1-measure values of 0.852 for the detection task and 0.828 when an exact match is required, which is better than the state-of-the-art results for temporal expression detection on TimeBank.

1 Introduction

Temporal information extraction in free text has been a research focus since 1995, when temporal expressions, sometimes also referred to as time expressions or TIMEXes, were processed as single capitalized tokens within the scope of the Message Understanding Conference (MUC) and the Named Entity Recognition task. As the demand for more deep semantic analysis tools increased, rule-based systems were proposed to solve this problem. The rule-based approach is characterized by providing decent results with a high precision level, and yields the rules, which can be easily interpreted by humans. With the advent of new annotated linguistic corpora, supervised machine-learning approaches have become the enabling technique for many problems in natural language processing, among which are named entity recognition, parsing, tagging and semantic role labeling. In 2004 the Automated Content Extraction (ACE) launched a competition campaign for Temporal Expression Recognition Normalization, TERN. The tasks were set to identify temporal expressions in free text and normalize them providing ISO-based date-time values.

While the ACE TERN initiative along with the provided corpus aimed at the recognition and normalization problems, more advanced temporal processing on the same dataset was not possible. The most recent annotation language for temporal expressions, TimeML [PCI03], and the underlying annotated corpus TimeBank [PHS03], opens up new horizons for automated temporal information extraction and reasoning.

A large number of rule-based and machine learning approaches were proposed for identification of temporal expressions. Comparative studies became possible with standardized annotated corpora, such as the ACE TERN and TimeBank. While the ACE TERN corpus is very often used for performance reporting, it restricts the temporal analysis to identification and normalization. By contrast, TimeBank provides a basis for all-around temporal processing, but lacks experimental results. In this paper we describe and compare two supervised machine learning approaches for identifying temporal information in free text. Both are trained on TimeBank, but follow two different classification techniques: token-by-token following B-I-O encoding and constituent-based classifications.

The remainder of the paper is organized as follows. In Section 2 we provide the details of relevant work done in this field along with corpora and annotations schemes used. Section 3 describes the approaches. Experimental setup, results and error analysis are provided in Section 4. Finally, Section 5 gives an outlook for further improvements and research.

2 Related Work

Since the task of temporal information extraction is not new, there have been a large number of implementations primary developed in the scope of the ACE TERN evaluations with the underlying corpus. As TimeBank provides annotations for more deep semantic analysis of temporal aspects in natural language, there have been also attempts of temporal taggers for this corpus, but very few. For better understanding of the performance levels provided in the paper we first describe evaluation metrics defined for the temporal expression recognition task and then the datasets and methods used in previous research.

2.1 Evaluation Metrics

With the start of the ACE TERN competition in 2004, two major evaluation conditions were proposed: Recognition+Normalization (full task) and Recognition only [TE04].

2.1.1 Detection (Recognition)

Detection is a preliminary task towards the full TERN task, in which temporally relevant expressions have to be found. The scoring is very generous and implies a minimal overlap in the extent of the reference and the system output tags. As long as there is at least one overlapping character, the tags will be aligned. Any alignment of the system output tags are scored as a correct detection.

2.1.2 Sloppy Span

Spans usually refer to strict match of both boundaries (the extent) of a temporal expression (see Exact Match). “Sloppy” admits recognized temporal expressions as long as their right-side boundary is the same as in the corresponding TimeBank’s extents [BA05]. The motivation was to assess the correctness of temporal expressions recognized in TimeBank, which was reported as inconsistent with respect to some left-side boundary items, such as determiners and pre-determiners.

2.1.3 Exact Match (Bracketing or Extent Recognition)

Exact match measures the ability of the system to correctly identify the extent of the TIMEX. The extent of the reference and the system output tags must match exactly the system output tag to be scored as correct.

2.2 Datasets

To date, there are two annotated corpora used for performance evaluations of temporal taggers, the ACE TERN corpus and TimeBank [PHS03]. In this section we provide a brief description of the temporal corpora and annotation standards.

2.2.1 ACE TERN Corpus

Most of the implementations referred to as the state-of-the-art were developed in the scope of the ACE TERN 2004. For evaluations, a training corpus of 862 documents with about 306 thousand words was provided. Each document represents a news article formatted in XML, in which TIMEX2 XML tags denote temporal expressions. The total number of temporal expressions for training is 8047 TIMEX2 tags with an average of 10.5 per document. The test set comprises 192 documents with 1828 TIMEX2 tags [Ferro04].

The annotation of temporal expressions in the ACE corpus was done with respect to the TIDES annotation guidelines [Ferro03]. The TIDES standard specifies so-called markable expressions, whose syntactic head must be an appropriate lexical trigger, e.g. “minute”, “afternoon”, “Monday”, “8:00”, “future” etc. When tagged, the full extent of the tag must correspond to one of the grammatical categories: nouns (NN, NNP), noun phrases (NP), adjectives (JJ), adjective phrases (ADJP), adverbs (RB) and adverb phrases (ADVP). According to this, all pre- and postmodifiers as well as dependent clauses are also included to the TIMEX2 extent, e.g. “five days after he came back”, “nearly four decades of experience”. Such a broad extent for annotations is of course necessary for correct normalization, but on the other hand, introduces difficulties for exact match. Another important characteristic of the TIDES standard are the nested temporal expressions as for example:

<TIMEX2>The<TIMEX2 VAL = "1994"> 1994 </TIMEX2> baseball season </TIMEX2>

2.2.2 TimeBank Corpus

The most recent annotation language for temporal expressions, TimeML [PCI03], with an underlying corpus TimeBank [PHS03], opens up new avenues for temporal information extraction. Besides the specification for temporal expressions, i.e. TIMEX3, which is to a large extent inherited from TIDES, TimeML provides a means to capture temporal semantics by annotations with suitably defined attributes for fine-grained specification of analytical detail [BP07]. The annotation schema establishes new entity and relation marking tags along with numerous attributes for them. This advancement influenced the extent for event-based temporal expression, in which dependent clauses are no longer included into TIMEX3 tags. The TimeBank corpus includes 186 documents with 68.5 thousand words and 1423 TIMEX3 tags.

2.3 Approaches for Temporal Tagging

As for any recognition problem, there are two major ways to solve it. Historically, rule-based systems were first implemented. Such systems are characterized by a great human effort in data analysis and rule writing. Delivering high precision results such systems can be successfully employed for recognition of temporal expressions, whereas the recall reflects the effort put into the rule development. By contrast, machine learning methods require an annotated training set, and with a decent feature design and a minimal human effort can provide comparable or even better results than rule-based implementations. As the temporal expression recognition is not only about to detect them but also to provide an exact match, machine learning approaches can be divided into token-by-token classification following B(egin)-I(nside)-O(utside) encoding and binary constituent-based classification, in which an entire chunk-phrase is under consideration to be classified as a temporal expression or not. In this case, exact segmentation is the responsibility of the chunker or the parser used.

2.3.1 Rule-based Systems

One of the first well-known implementations of temporal taggers was presented in [Mani00]. The approach relies on a set of hand-crafted and machine-discovered rules, which based upon shallow lexical features. On average the system achieved a value of 0.832 for F1-measure against hand-annotated data. The dataset used comprised a set of 22 New York Times articles and 199 transcripts of Voice of America taken from the TDT2 collection [Graff99]. It should be noted that the reported performance was provided in terms of an exact match. Another example of rule-based temporal taggers is Chronos, described in [NM04], which achieved the highest F1-scores in the ACE TERN 2004 of 0.926 and 0.878 for recognition and exact match respectively.

Recognition of temporal expressions using TimeBank as an annotated corpus, is reported in [BA05] and based on a cascaded finite-state grammar (500 stages and 16000 transitions). A complex approach achieved an F1-measure value of 0.817 for exact match and 0.896 for detecting "sloppy" spans. Another known implementation for TimeBank is GUTime[1] – an adaptation of [Mani00] from TIMEX2 to TIMEX3 with no reported performance level.

2.3.2 Machine Learning Recognition Systems

Successful machine learning TIMEX2 recognition systems are described in [Ahn05; HCD05; PST07]. Proposed approaches made use of a token-by-token classification for temporal expressions represented by B-I-O encoding with a set of lexical and syntactic features, e.g., token itself, part-of-speech tag, label in the chunk phrase and the same features for each token in the context window. The performance levels are presented in Table 1. All the results were obtained on the ACE TERN dataset.

Approach	F1 (detection)	F1 (exact match)
Ahn et al. [Ahn05]	0.914	0.798
Hacioglu et al. [HCD05]	0.935	0.878
Poveda et al. [PST07]	0.986	0.757

Table 1: Performance of machine learning approaches with B-I-O encoding

Constituent-based, also known as chunk-based, classification approach for temporal expression recognition was presented in [Ahn07]. By comparing to the previous work of the same authors [Ahn05] and on the same ACE TERN dataset, the method demonstrates a slight decrease in detection with F1-measure of 0.844 and a nearly equivalent F1-measure value for exact match of 0.787.

The major characteristic of machine learning approaches was a simple system design with a minimal human effort. Machine-learning based recognition systems have proven to have a comparable recognition performance level to state-of-the-art rule-based detectors.

[1] http://www.timeml.org/site/tarsqi/modules/gutime/index.html

3 Our Approaches

The approaches presented in this section employ a supervised machine learning algorithm following a similar feature design but different classification strategies. Both classifiers implement a Maximum Entropy Model[2].

3.1 Token-based Classification Approach

Multi-class classifications, such as the one with B-I-O encoding, are a traditional way for detection tasks in natural language processing, for example Named Entity Recognition and chunking. This method does not require deep and time-consuming pre-processing and merely relies on shallow lexical and syntactical features generated for each token under consideration and in the context window. For this approach we employ the OpenNLP toolkit[3] when pre-processing the data. The toolkit makes use of the same Maximum Entropy model for detecting sentence boundaries, part-of-speech (POS) tagging and parsing tasks [Ratn96; Ratn99]. The tokenized output along with detected POS tags is used for generating feature vectors with one of the labels from the B-I-O encoding. The feature-vector design comprises the initial token in lowercase, POS tagger, character type and character type pattern[4]. Character type and character type pattern features are implemented following Ahn et al. [Ahn05]. The patterns are defined by using the symbols X, x and 9. X and x are used for character type as well as for character type patterns for representing capital and lower-case letters for a token. 9 is used for representing numeric tokens. Once the character types are computed, the corresponding character patterns are produced. A pattern consists of the same symbols as character types, and contains no sequential redundant occurrences of the same symbol. For example, the token “January” has character type “Xxxxxxx” and pattern “X(x)”. The same feature design is applied to each token in the context window of three tokens to the left and to the right in the sequence limited by sentence boundaries.

[2] http://maxent.sourceforge.net/
[3] http://opennlp.sourceforge.net/
[4] In literature such patterns are also known as shorttypes.

3.2 Constituent-based Classification Approach

For constituent-based classification the entire phrase is under consideration to be labeled as a TIMEX or not. We restrict the classification for the following phrase types and grammatical categories derived from the Penn Treebank tagset: nouns (NN), proper nouns (NNP), cardinals (CD), noun phrases (NP), adjectives (JJ), adjective phrases (ADJP), adverbs (RB), adverbial phrases (ADVP) and prepositional phrases (PP). In order to make it possible, for each sentence we parse the initial input line with a Maximum Entropy parser [Ratn99] and extract all phrase candidates with respect the types defined above. Each phrase candidate is examined against the manual annotations for temporal expressions found in the sentence. Those phrases, which correspond to the temporal expressions in the sentence are taken as positive examples, while the rest are considered as a negative set. Only one sub-tree from a parse is marked as positive for a distinct TIMEX at once. After that, for each candidate we produce a feature vector, which includes the following features: head phrase, head word, part-of-speech for head word, character type and character type pattern (see Section 3.1) for head word as well as for the entire phrase. For example, the constituent "January 30th" has character type "Xxxxxxx 99xx" and pattern "X(x) (9)(x)".

4 Experiments, Results and Error Analysis

All experiments were conducted following 10-fold cross validation and evaluated with respect to the to the TERN 2004 evaluation plan described in Section 2.1.

4.1 Token-based Classification Experiments

After pre-processing the textual part of TimeBank, we received a set of 26509 tokens with 1222 correctly aligned TIMEX3 tags. Due to a token-based classification strategy our evaluation for the detection task is stricter than the ACE TERN methodology (see Section 2.1). While the ACE TERN evaluations measure the results requiring one overlapping character to be correct we compare obtained labels for entire tokens. The experimental results demonstrated the performance in detection of temporal expressions with precision, recall and F1-measure at 0.928, 0.628 and 0.747 respectively. When an exact match is required, the classifier performs at the level of 0.888, 0.382 and 0.532 for precision, recall and F1-measure respectively.

4.2 Constituent-based Classification Experiments

After pre-processing the TimeBank corpus of 182 documents we had 2612 parsed sentences with 1224 temporal expressions in them. 2612 sentences resulted in 49656 phrase candidates.

After running experiments the classifier demonstrated the performance in detection of TIMEX3 tags with precision, recall and F1-measure at 0.872, 0.836 and 0.852 respectively. Since the candidate phrases provided by the parser do not always exactly align annotated temporal expressions, the results for the exact match experiments are constrained by an estimated upper-bound recall of 0.919. The experiments on exact match demonstrated a small decline of performance level and received scores of 0.866, 0.796 and 0.828 for precision, recall and F1-measure respectively.

4.3 Comparison and Improvements

Comparing the performance levels of the tested temporal taggers, we discovered the differences in classification results of chunk-based and token-based approaches with corresponding F1-measure values of 0.852 vs. 0.747 for detection, and 0.828 vs. 0.532 for exact match. Previous experiments on the ACE TERN corpus, especially those in [Ahn05; Ahn07], confirmed the same phenomenon and reported a drop in F1-measure between detection and exact match, but the token-based approach delivers generally better results. For our experimental results we assume that the problem lies in a local token classification with pure lexico-syntactic features. A context-dependent classification may solve it. In order to prove this hypothesis, the next series of experiments is performed with an additional feature set, which contains the classification results obtained for preceding tokens, so called Maximum Entropy Markov Model. The experimental setup varies the number of previously consecutive obtained labels between 1 and 3 with the same context window size of 3 tokens to the left and to the right. The context is considered within the sentence only. The results of these experiments are presented in Table 2. The number of the previously obtained labels used as features is denoted by N, with N=0 as a baseline, which is described above (see Section 3.1)

N	Detection			Exact match		
	P	R	F1	P	R	F1
0	0.928	0.628	0.747	0.888	0.382	0.532
1	0.946	0.686	0.793	0.921	0.446	0.599
2	0.94	0.652	0.768	0.911	0.426	0.578
3	0.936	0.645	0.762	0.905	0.414	0.566

Table 2: Performance of machine learning approaches with B-I-O encoding

It is worth to mention that by taking into account labels obtained for preceding tokens the performance level rises and reaches the maximum at N=1 for both, the detection and exact match tasks, and decreases from N=2 onwards.

Putting the received figures in context, we can conclude that the chunk-based machine learning approach for temporal expression recognition performed at a comparable operational level to the state-of-the-art rule-based approach of Boguraev and Ando [BA05] and outperformed it in exact match. A comparative performance summary is presented in Table 3.

	P	R	F1
Detection			
CBC approach	0.872	0.836	0.852
Sloppy Span			
Boguraev and Ando [BA05]	0.852	0.952	0.896
Exact Match			
CBC approach	0.866	0.796	0.828
Boguraev and Ando [BA05]	0.776	0.861	0.817

Table 3. Comparative performance summary for the constituent-based classification (CBC) approach.

4.4 Error Analysis

Analyzing the classification errors we see several causes for them. We realized that the current version of TimeBank, TimeBank 1.2, is still noisy with respect to annotated data. An ambiguous use of temporal triggers in different context, like “today”, “now”, “future”, makes correct identification of relatively simple temporal expressions difficult. Sometimes it is very hard even for humans to identify the use of obvious temporal triggers in a specific context. As a result, many occurrences of such triggers remained unannotated, for which TIMEX3 identification could not be properly carried out. Apart of obvious incorrect parses, inexact alignment between temporal expressions and candidate phrases was caused by annotations that occurred at the middle of a phrase, for example “eight-years-long”, “overnight”, “yesterday’s”. In total there are 99 TIMEX3 tags (or 8.1%) misaligned with the parser output, which resulted in 53 (or 4.3%) undetected TIMEX3s. Definite and indefinite articles are unsystematically left out or included into TIMEX3 extent, which introduces an additional bias in classification.

5 Conclusion and Future Work

In this paper we presented two machine learning approaches for detecting temporal expressions using a recent annotated corpus for temporal information, TimeBank. The first approach implements a token-by-token classifier following B-I-O encoding, the second one performs a constituent-based classification. The feature design for both methods is very similar and takes into account contentual and contextual features. The obtained results were evaluated with respect to the ACE TERN evaluation plan for the two following tasks: detection and exact match. As the evaluation showed, both approaches provide a good performance level for detection temporal expressions, whereas constituent-based classification outperforms token-based one, with F1-measure values of 0.852 vs. 0.747. If an exact match is required, only the constituent-based classification can provide reliable recognition with a F1-measure value of 0.828. For the same task token-based classification reaches only 0.532 in terms of F1-measure. The token-based method in this case has very low recall values, which results in a low overall performance. By employing additional features that represent the classification history of previous tokens, so called Maximum Entropy Markov Model, the method increases the performance level and reaches its maximum, when only the classification result for the previous token is used (with F1-measures of 0.793 and 0.599 for detection and exact match respectively).

Our best results were obtained by the binary constituent-based classification approach with shallow syntactic and lexical features. The method achieved a performance level to a rule-based approach presented in [BA05] and for the exact match task our approach even outperforms the latter. Although a direct comparison with other state-of-the-art systems is not possible, due to different evaluation corpora, annotation standards and corpus volumes, our experiments disclose a very important characteristic. While the recognition systems in the ACE TERN 2004 reported a substantial drop of F1-measure between detection and exact match results (6.5 - 11.6%), our phrase-based detector demonstrates a light decrease in F1-measure (2.4%), whereas the precision declines only by 0.6%. This important finding leads us to the conclusion that most of TIMEX3s in TimeBank can be detected at a phrase-based level with a reasonably high performance.

Despite a good recognition performance level there is, of course, room for improvement. Many implementations in the ACE TERN 2004 employ a set of apparent temporal tokens as one of the features; by contrast, we learn them from data. In our implementation, the classifier has difficulties with very simple temporal expressions such as “now”, “future”, “current”, “currently”, “recent”, “recently”. A direct employment of vocabularies with temporal tokens may substantially increase the F1-measure of the presented methods, however, it yet has to be proven. As reported in [Ahn07] a precise recognition of temporal expressions is a prerequisite for accurate normalization.

With our detector and a future normalizer we are able make the first step towards a comprehensive temporal analysis of free text. Our future work will be focused on improving current results by a new feature design, finalizing the normalization task and identification of temporal relations. All these components will result in a solid system infrastructure for all-around temporal analysis.

Acknowledgments

This work has been partly funded by the Flemish government (through IWT) and by Space Applications Services NV as part of the ITEA2 project LINDO (ITEA2-06011).

References

[Ahn05] Ahn, D.; Adafre, S. F.; de Rijke, M.: Extracting Temporal Information from Open Domain Text: A Comparative Exploration. Digital Information Management, 3(1):14—20, 2005.

[Ahn07] Ahn, D.; van Rantwijk, J.; de Rijke, M.: A Cascaded Machine Learning Approach to Interpreting Temporal Expressions. In: NAACL-HLT 2007, 2007.

[BA05] Boguraev, B.; Ando, R. K.: TimeBank-Driven TimeML Analysis. In: Annotating, Extracting and Reasoning about Time and Events. Dagstuhl Seminar Proceedings. Dagstuhl, Germany, 2005.

[BP07] Boguraev, B.; Pustejovsky, J.; Ando, R.; Verhagen, M.: TimeBank Evolution as a Community Resource for TimeML Parsing. Language Resource and Evaluation, 41(1), 91—115, 2007.

[Ferro03] Ferro, L.; Gerber, L.; Mani, I.; Sundheim, B.; Wilson, G.: TIDES 2003 Standard for the Annotation of Temporal Expressions, http://timex2.mitre.org, 2003.

[Ferro04] Ferro, L.: TERN Evaluation Task Overview and Corpus, http://fofoca.mitre.org/tern_2004/ferro1_TERN2004_task_corpus.pdf, 2004

[Graff99] Graff, D.; Cieri, C.; Strassel, S.; Martey, N.: The TDT-2 Text and Speech Corpus. In: Proceedings of the DARPA Broadcast News Workshop, 1999; pp. 57—60.

[HCD05] Hacioglu, K.; Chen, Y.; Douglas, B.: Automatic Time Expression Labeling for English and Chinese Text. In: Proceedings of the Conference on Intelligent Text Processing and Computational Linguistics 2005. Lecture Notes in Computer Science, vol. 3406, Springer-Verlag, 2005; pp. 348—359.

[Mani00] Mani, I.; Wilson, G.: Robust Temporal Processing of News. In: Proceedings of the 38th Annual Meeting on Association for Computational Linguistics (Hong Kong, October 03 - 06, 2000). Annual Meeting of the ACL. Association for Computational Linguistics, Morristown, NJ, 2000; pp. 69—76.

[NM04] Negri, M.; Marseglia, L.: Recognition and Normalization of Time Expressions: ITC-irst at TERN 2004. Technical Report, ITC-irst, Trento, 2004.

[PST07] Poveda, J.; Surdeanu, M.; Turmo, J.: A Comparison of Statistical and Rule-Induction Learners for Automatic Tagging of Time Expressions in English. In: Proceedings of the International Symposium on Temporal Representation and Reasoning, 2007; pp. 141—149.

[PCI03] Pustejovsky, J.; Castaño, J.; Ingria, R.; Saurí, R.; Gaizauskas, R.; Setzer, A.; Katz, G.: TimeML: Robust Specification of Event and Temporal Expressions in Text. In: IWCS-5, Fifth International Workshop on Computational Semantics, 2003.

[PHS03] Pustejovsky, J.; Hanks, P.; Saurí, R.; See, A.; Day, D.; Ferro, L.; Gaizauskas, R.; Lazo, M.; Setzer, A.; Sundheim, B.: The TimeBank Corpus. Corpus Linguistics 2003, 647—656, 2003.

[Ratn96] Ratnaparkhi, A.: A Maximum Entropy Model for Part-of-Speech Tagging, In: Conference on Empirical Methods in Natural Language Processing, 1996; pp. 133—142.

[Ratn99] Ratnaparkhi, A.: Learning to Parse Natural Language with Maximum Entropy Models. Machine Learning, 34(1): 151—175, 1999.

[TE04] TERN 2004 Evaluation Plan, http://fofoca.mitre.org/tern_2004/tern_evalplan-2004.29apr04.pdf, 2004.