Extracting Temporal Information from Portuguese Texts

Francisco Costa and António Branco

University of Lisbon
fcosta@di.fc.ul.pt
Antonio.Branco@di.fc.ul.pt

Abstract. This paper reports on experimenting with the extraction of temporal information from Portuguese texts and presents LX-TimeAnalyzer, a tool that annotates a text with the temporal information conveyed by it. This tool is the first of its kind being reported for Portuguese, and its performance is similar to the state-of-the-art for other languages.

1 Introduction and Related Work

Extracting the temporal information present in a text is relevant to many Natural Language Processing applications, including question-answering, information extraction, and even document summarization, as summaries may be more readable if the information is presented in chronological order.

The two recent TempEval challenges [9, 10] focused on extracting the temporal information conveyed in written text and provided data that can be used to develop and evaluate systems that can automatically annotate a natural language text with the temporal information conveyed in it. Figure 1 shows an example of similarly annotated data.

```
<s>Em Washington, <TIMEX3 tid="t53" type="DATE"
value="1998-01-14">hoje</TIMEX3>, a Federal Aviation Administration <EVENT
eid="e1" class="OCCURRENCE" stem="publicar" aspect="NONE" tense="PPI"
polarity="POS" pos="VERB">publicou</EVENT> gravações do controlo de tráfego
aéreo da <TIMEX3 tid="t54" type="TIME" value="1998-XX-XXTNI">noite</TIMEX3>
em que o voo TWA800 <EVENT eid="e2" class="OCCURRENCE" stem="cair"
aspect="NONE" tense="PPI" polarity="POS" pos="VERB">caiu</EVENT>.</s>
<TLINK lid="l1" relType="BEFORE" eventID="e2" relatedToTime="t53"/>
<TLINK lid="l2" relType="OVERLAP" eventID="e2" relatedToTime="t54"/>
```

Fig. 1. Sample of Portuguese data with temporal annotations, corresponding to the fragment: *Em Washington, hoje, a Federal Aviation Administration publicou gravações do controlo de tráfego aéreo da noite em que o voo TWA800 caiu.*
The English equivalent is: *In Washington today, the Federal Aviation Administration released air traffic control tapes from the night the TWA Flight eight hundred went down.*

Terms denoting events, such as the event of releasing the tapes that is described in that text, are annotated using `EVENT` tags, and temporal expressions, such as *today*, are enclosed in `TIMEX3` tags. The attribute `value` of time expressions holds a normalized representation of the date or time they refer to (e.g. the word *today* denotes the date `1998-01-14` in this example). The `TLINK` elements at the end describe temporal relations between events and temporal expressions. For instance, the event of the plane going down is annotated as temporally preceding the date denoted by the temporal expression *today*.

The first TempEval challenge focused solely on the temporal relations. TempEval-2 additionally included tasks related to the identification and normalization of event terms and temporal expressions. Identification is concerned with classifying words in a text as to whether they are event terms or part of temporal expressions or none of these. Normalization is about determining the value of the various attributes of `EVENT` and `TIMEX3` elements, specially the `value` attribute of `TIMEX3` elements. By combining the outcome of all these tasks, it is possible to fully annotate raw text with temporal information (event terms, temporal expressions and temporal relations) in a way similar to what is shown in the example above. Table 1 shows the scores obtained by the best participant for each of these problems. The evaluation measures used were the f-measure for the problems of identifying the extents of event and time expressions and accuracy for the tasks dealing with the attributes. Full details can be found in [10].

Temporal expressions			Events		
Task	English	Spanish	Task	English	Spanish
Extents	0.86	0.91	Extents	0.83	0.88
`type`	0.98	0.99	`class`	0.79	0.66
`value`	0.85	0.83	`tense`	0.92	0.96
			`aspect`	0.98	0.89
			`polarity`	0.99	0.92

Table 1. Best system results for the various tasks of TempEval-2, according to [10].

2 Approach and Evaluation

The data that was used for the first TempEval has recently been adapted to Portuguese, as reported in [3]. The documents that make up this corpus were translated to Portuguese, and the annotations adapted to the language. The fragment presented above in Figure 1 is taken from this corpus. The training subset contains 68,351 words, 6,790 events, 1,244 temporal expressions and 5,781 temporal relations.

These data allow for the training and evaluation of temporal processing systems for Portuguese. In Table 2 we include information about the performance

of our system LX-TimeAnalyzer, evaluating each subtask that was evaluated in TempEval-2 (with the exception of temporal relation classification, which is reported in [2, 4]). We use the same evaluation measures as in TempEval-2 (f-measure for extent identification and accuracy for the tasks dealing with the attributes). It must be noted that: (i) the Portuguese data are an adaptation of the English data used in the first TempEval, (ii) the results in Table 1 refer to TempEval-2, (iii) the English data of TempEval and TempEval-2 are not identical, although there is a large overlap between them. For the data of the first TempEval there are unfortunately no published results that we know of concerning the identification and normalization of temporal expressions and event terms, as TempEval-1 focused only on temporal relations. It is thus important to note that our results are fully not comparable to the results for English (and they are even less comparable to the results for Spanish, as those are based on completely different data).

Temporal expressions		Events	
Task	Score	Task	Score
Extents	0.85	Extents	0.72
`type`	0.91	`class`	0.74
`value`	0.81	`tense`	0.95
		`aspect`	0.96
		`polarity`	0.99

Table 2. Evaluation of LX-TimeAnalyzer on the test data

The document to be processed is initially tagged with a morphological analyzer [1]. This tool annotates each word with its part-of-speech category (noun, verb, etc.), its lemma (i.e. its dictionary form), and a tag describing its inflection features.

For the tasks we addressed via machine learning techniques, we employed Weka's [11] implementation of the C4.5 algorithm, using the training data for training and the held-out test data for evaluation.

2.1 Event Identification and Normalization

A simple solution to identifying event terms in text is to classify each word as to whether it denotes an event or not. This strategy is not very efficient, since (i) some very frequent words cannot possibly denote events (e.g. determiners, conjunctions etc.), and (ii) most event terms are verbs or nouns (92% according to the training data). Nevertheless, for the sake of reproducibility, we followed this straightforward approach.

The classifier features employed are:

– **Features about the last characters of the lemma**
A Boolean attribute represents whether the lemma ends in one of several

word endings from a hand-crafted list. This list includes suffixes such as *-mento*. The motivation is that this information may be useful especially to separate eventive nouns from non-eventive nouns. There are additional attributes that include information about the last two characters of the lemma and the last three characters of the lemma; they are intended to capture suffixes not covered by the list of suffixes.

- **The part-of-speech and the inflection tag assigned by the tagger.** As argued above, information about part-of-speech can rule out many words in a document. The inflection tag may also be relevant. For instance, even though singular forms are more common than plural forms for both eventive and non-eventive nouns, this difference is sharper in the case of eventive nouns (since these denote multiple or repeated events).
- **The part-of-speech and the inflection tag of the preceding word token, the following word token, the preceding word token bigram, the following word token bigram**.
  These attributes are used in order to capture some contextual information.
- **Whether the preceding token was classified as an event**
  The intuition is that adjacent event terms are infrequent.

Our result for this task (0.72 f-measure) is worse than the best systems of TempEval-2 for both English (0.83) and Spanish (0.88).

We believe that the major cause of this differences is that these systems used features based on WordNet, which we were unable to experiment with as there is no available WordNet for Portuguese verbs.

The task of event normalization is concerned with the annotation of the several attributes appropriate for `<EVENT>` elements. The values of many of the attributes of `<EVENT>` elements are already provided by the morphological analyzer: `stem` (the term's dictionary form), `tense` (tense) and `pos` (part-of-speech). Three attributes are not, however: `aspect`, `polarity` and `class`.

For the `polarity` attribute, we simply check whether one the three preceding words is a negative word—*não* "not", *nunca* "never", *ninguém* "nobody", *nada* "nothing", *nenhum/nenhuma/nenhuns/nenhumas* "no, none", *nenhures* "nowhere"— and there is no other event intervening between this n-word and the event that is being annotated. The accuracy for this heuristic is 0.99, considering all annotated events in both the training and the test data. On the test data, the accuracy of this simple heuristic is also 0.99, which is identical to the best score in TempEval-2 for English (0.99) and better than the one for Spanish (0.92).

In the Portuguese data, the attribute `aspect` only encodes whether the verb form is part of a progressive construction. This attribute is also computed symbolically, and the implementation simply checks for gerund forms (e.g. *fazendo*) or constructions involving an infinite verb form immediately preceded by the preposition *a* (*a fazer*). Once again considering all the data (both training and testing data), this approach has an accuracy of 0.99. On the evaluation data, its accuracy is 0.96, in between the TempEval-2 best scores for English (0.98) and Spanish (0.89).

The most interesting and hardest problem of event normalization is determining the value of the `class` attribute of `<EVENT>` elements. This attribute includes some information about the semantic class of event terms, distinguishing `REPORTING`, `PERCEPTION` and `ASPECTUAL` terms from the others, and also includes some aspectual distinctions in the spirit of [8, 5], distinguishing `STATE` situations from non-stative events, marked as `OCCURRENCE`s. It is thus sensitive to both lexical and contextual (i.e. syntactic) information. For this attribute, a specific classifier was trained, with a very limited set of features:

- **The lemma of the event term being classified**
  This type of information is highly lexicalized, so it is expected that the lemma of the word token can be quite informative.
- **Contextual features**
  These attributes encode the part-of-speech of the previous word and that of the next word, and the following bigram of inflection tags.

We experimented with more features, similar to the ones used for event detection, but they did not improve the results. We obtained a result of 0.74.

2.2 Temporal Expression Identification and Normalization

In order to identify temporal expressions, we trained a classifier that, to each word in the text, assigns one of three labels: `B` (begin), `I` (inside), `O` (outside). The features employed were:

- **Features about the current token**
  These include the token's part-of-speech and its inflection tag. Additionally, there is an attribute that checks whether the current token's lemma is part of a list of temporal adverbs. This is specially useful for the `B` class, which is the one with the highest error rate.
- **Features about the previous token and the following one**
  These features are taken from the morphological analyzer and encode part-of-speech and inflection tag.
- **The classification for the previous token**
  Tokens classified as `I` cannot directly follow tokens classified as `O`.
- **Whether there is white space before the current token and the previous one**
  The reason behind this attribute is to treat punctuation and special symbols in a special manner (they are tokenized separately; e.g. a time expression of the form `XXXX-XX-XX` is tokenized into five word tokens).
- **Whether (i) the current token's lemma was seen in the training data at the beginning of a temporal expression, or (ii) it was seen inside a temporal expression, or (iii) the bigram of lemmas formed by the current token's lemma and the next one's was seen inside a temporal expression**
  Instead of using an attribute encoding the lemma directly, we used a series of Boolean attributes capturing distinctions that are expected to help classification.

As shown in Table 2, this component shows an f-measure of 0.85 for the `B` and `I` classes.

The task of temporal expression normalization consists in identifying the value of the `TIMEX3` attributes `type` and `value`. LX-TimeAnalyzer solves it symbolically. The normalization rules take as input the following parameters:

- The word tokens composing the temporal expression, and their morphological annotation
- The document's creation time
- An anchor. This is another temporal expression that is often required for normalization. An expression like *the following day* can only be normalized if its anchor is known. We use the previous temporal expression that occurs in the same text and that is not a duration, a simple heuristic similar to previous approaches found in the literature.
- The broad tense (*present*, *past*, or *future*) of the closest verb in the sentence where it occurs, with the distance being measured in number of word tokens from either boundary of the time expression. For example, all past tenses are treated as *past*. This is used to decide whether an expression like *February* refers to the previous or the following month of February (relative to the document's creation time).

These rules are implemented by a Java method. It takes approximately 1600 lines of code and is recursive: e.g. when normalizing an expression like *terça de manhã* "Tuesday morning", the expression *terça* "Tuesday" is normalized first, and then its normalized `value` is changed by appending `TMO` (with `T` being the time separator and `MO` the way to represent the vague expression "morning"); its `type` is also changed from `DATE` to `TIME`. The same method fills in both the `value` and the `type` attributes of `TIMEX3` elements. This implementation was conducted by looking at the examples in the training data, and additionally at a small set (c. 5000 words) of news reports taken from on-line newspapers.

The accuracy of LX-TimeAnalyzer at predicting the value of the `value` attribute of `TIMEX3` elements is 0.81 on the test data. For the `type` attribute this is 0.91.

3 Concluding Remarks

Full temporal information processing is fairly recent. Only in the TempEval-2 challenge, last year in 2010, were there systems capable of fully annotating raw text with temporal information (e.g. [7, 6]).

LX-TimeAnalyzer is the first fully-fledged temporal analyzer for Portuguese. It performs in line with the state-of-the-art for other languages, although (i) the data used for evaluation are not fully comparable, and (ii) event detection is somewhat worse, but can possibly be improved by incorporating information similar to that in WordNet.

References

1. Branco, A., Silva, J.: A suite of shallow processing tools for portuguese: LX-Suite. In: Proceedings of the 11^{th} Conference of the European Chapter of the Association for Computational Linguistics (EACL'06). Trento, Italy (2006)
2. Costa, F.: Processing Temporal Information in Unstructured Documents. Ph.D. thesis, Universidade de Lisboa, Lisbon (to appear)
3. Costa, F., Branco, A.: Temporal information processing of a new language: Fast porting with minimal resources. In: ACL2010—Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (2010)
4. Costa, F., Branco, A.: LX-TimeAnalyzer: A temporal information processing system for Portuguese. Tech. rep., Universidade de Lisboa, Faculdade de Ciências, Departamento de Informática (to appear)
5. Dowty, D.R.: Word Meaning and Montague Grammar: the Semantics of Verbs and Times in Generative Semantics and Montague's PTQ. Reidel, Dordrecht (1979)
6. Llorens, H., Saquete, E., Navarro, B.: TIPSem (English and Spanish): Evaluating CRFs and semantic roles in TempEval-2. In: Erk, K., Strapparava, C. (eds.) SemEval 2010—5^{th} International Workshop on Semantic Evaluation—Proceedings of the Workshop. pp. 284–291. Uppsala University, Uppsala, Sweden (2010)
7. UzZaman, N., Allen, J.F.: TRIPS and TRIOS System for TempEval-2: Extracting temporal information from text. In: Erk, K., Strapparava, C. (eds.) SemEval 2010—5^{th} International Workshop on Semantic Evaluation—Proceedings of the Workshop. pp. 276–283. Uppsala University, Uppsala, Sweden (2010)
8. Vendler, Z.: Verbs and times. Linguistics in Philosophy pp. 97–121 (1967)
9. Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Pustejovsky, J.: SemEval-2007 Task 15: TempEval temporal relation identification. In: Proceedings of SemEval-2007 (2007)
10. Verhagen, M., Saurí, R., Caselli, T., Pustejovsky, J.: SemEval-2010 task 13: TempEval-2. In: Strapparava, C., Erk, K. (eds.) SemEval 2010—5^{th} International Workshop on Semantic Evaluation—Proceedings of the Workshop. pp. 51–62. Uppsala University, Uppsala, Sweden (2010)
11. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann, San Francisco (2005), second edition