Kronos-it: a Dataset for the Italian Semantic Change Detection Task

Pierpaolo Basile
University of Bari A. Moro
Dept. Computer Science
E. Orabona 4, Italy
pierpaolo.basile@uniba.it

Giovanni Semeraro
University of Bari A. Moro
Dept. Computer Science
E. Orabona 4, Italy
giovanni.semeraro@uniba.it

Annalina Caputo
ADAPT Centre
Dublin City University
Dublin, Ireland
annalina.caputo@dcu.ie

Abstract

This paper introduces Kronos-it, a dataset for the evaluation of semantic change point detection algorithms for the Italian language. The dataset is automatically built by using a web scraping strategy. We provide a detailed description about the dataset and its generation, and four state-of-the-art approaches for the semantic change point detection are benchmarked by exploiting the Italian Google n-grams corpus.

1 Background and Motivation

Computational approaches to the problem of language change have been gaining momentum over the last decade. The availability of long-term and large-scale digital corpora, and the effectiveness of methods for representing words over time, are the prerequisite behind this interest. However, only few attempts have focused on the evaluation, due to two main issues. First, the amount of data involved limits the possibility to perform a manual evaluation and, secondly, to date no open dataset for the diachronic semantic change has been made available. This last issue has roots in the difficulties of building a gold-standard for detecting the semantic change of terms in a specific corpus or language. The result is a fragmented set of data and evaluation protocols, since each work in this area has used different evaluation datasets or metrics. This phenomenon can be gauged from (Tahmasebi et al., 2019), where it is possible to count at least twenty different datasets used for the evaluation. In this paper, we describe how to build a dataset for the evaluation of semantic change point detection algorithms. In particular, we adopt a web scraping strategy for extracting information from an online Italian dictionary. The goal of the extraction is to build a list of lemmas with a set of change points for each lemma. The change points are extracted by analysing information about the year in which the lemma with a specific meaning is observed for the first time. Relying on this information we build a dataset for the Italian language that can be used to evaluate algorithms for the semantic change point detection. We provide a case study in which four different approaches are analysed using a unique corpus.

The rest of the article is organised as follows: Section 2 describes how our dataset is built, while Section 3 provides details about the approaches under analysis and the evaluation. Finally, Section 4 closes the paper and provides possible future work.

2 Dataset Construction

The main goal of the dataset is to provide for each lemma a set of years which indicate a semantic change for that lemma. Some dictionaries provide historical information about meanings, for example the year in which each meaning is observed for the first time. The main problem is that generally these dictionaries are not digitally available or they are in a format that is not machine readable.

Regarding the Italian language, the dictionary "Sabatini Coletti"[1] is available on-line. It provides for some lemmas the year in which each meaning was observed for the first time. For example, taking into account the entry for the word "imbarcata" from the dictionary, we capture its original meaning "Group of people who gather to find each other, to leave together", and other two meanings: 1) "Acrobatic manoeuvre of an air-plane" introduced in 1929; and 2) "fall in love" introduced in 1972.

[1] https://dizionari.corriere.it/dizionario_italiano/

We setup a web scraping algorithm able to extract this information from the dictionary. In particular, the extraction process is composed of several steps:

1. Downloading the list of all lemmas occurring in the online dictionary with the corresponding URL. We obtain a list of 34,504 lemmas;

2. For each lemma, extracting the section of the web page containing the definition with the list of all possible meanings. We obtain a final list of 34,446 definitions;

3. For each definition, extracting the year in which that meaning was introduced. For a given lemma, we are not able to assign the correct year to each of its meaning, but we can only extract a year associated with the lemma. This happens because the dictionary does not follow a clear template for assigning the year to each meaning. Although associating the year of change to the definition of the meaning is not useful for the purpose of our evaluation, it could help to understand the reason behind the semantic change. We plan to fix this limitation in a further release of the dataset. In the rest of the paper we call change point (CP) each pair (lemma, year);

4. Removing those change points that are expressed in the form "III sec." (*III century*) because they refer to a broad period of time rather than to a specific year.

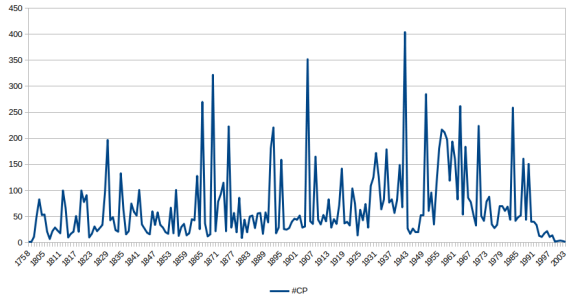

Figure 1: The distribution of change points over time.

The final dataset[2] contains 13,818 lemmas and 13,932 change points. The average change points for lemma is 1.0083 with a standard deviation of 0.0924. The maximum number of change points for lemma is 3 and the number of lemmas with more than one change point is 113. The oldest reported change point is 1758, while the most recent one is 2003; this suggests that the dictionary is outdated and it does not contain more recent meanings.

The dataset is provided in textual format and reports for each row the lemma followed by a list of years, each one representing a change point. For example:

```
enzima 1892
monopolistico 1972
tamponare 1886 1950
elettroforesi 1931
fuoricorso 1934
```

The low number of change points for lemma reflects the fact that generally, the first meaning has no information about the year it first appeared in or that its time period is expressed in the form of century. This means that all the other meanings are additional meanings introduced after the main one. However, there are some more recent words for which the first year associated with that entry corresponds to the year in which the word is observed for the first time. Unfortunately, it is not easy to automatically discern the two cases.

Finally, we report the distribution of change points over time in Figure 1. The years with a peak are 1942, 1905 and 1869 with respectively 404, 352 and 322 change points.

3 Evaluation

For the evaluation we adopt our dataset as gold-standard and the Italian Google n-grams (Michel et al., 2011) as corpus[3].

Google n-grams provides n-grams extracted from the Google Books project. The corpus is composed of several compressed files. Each file contains tab-separated data, each line has the following format: *ngram TAB year TAB match_count TAB volume_count NEWLINE*. For example:

```
parlare di pace e di 2005 4 4
parlare di pace e di 2006 3 3
parlare di pace e di 2007 7 7
parlare di pace e di 2008 2 2
parlare di pace e di 2009 4 4
```

The first line tells us that in 2005, the 5-grams *"parlare di pace e di"* occurred 4 times overall, in 4 distinct books.

[2] https://github.com/pippokill/kronos-it

[3] http://storage.googleapis.com/books/ngrams/books/datasetsv2.html

In particular, we use the 5-grams corpus and we limit the analysis to words that occur at least in twenty 5-grams. Moreover, we lowercase words and filter out all words that do not match the following regular expression: *[a-zéèàìòù]+*. We limit our analysis to the period [1900-2012].

In order to build the context words by using 5-grams, we adopt the technique described in (Ginter and Kanerva, 2014). Given a 5-gram (w_1, w_2, w_3, w_4, w_5), it is possible to build eight pairs: (w_1, w_2) (w_1, w_3) ... (w_1, w_5) and (w_5, w_1) (w_5, w_2) ... (w_5, w_4). Then, for each pair (w_i, w_j), a sliding window method also visits (w_j, w_i) by obtaining 16 training examples from each 5-gram.

We investigate four systems for representing words over time and then we apply a strategy for extracting change points from each technique. Finally, we evaluate the accuracy of each approach by using our dataset as gold standard.

3.1 Representing words over time

We adopt four techniques for representing words over time. The first strategy is based only on word co-occurrences, the other three exploit Distribution Semantic Models (DSM). In particular, the techniques are:

Collocation. This approach is very simple and it is used as baseline. The idea is to extract for each word and each time period the set of relevant collocations. A collocation is a sequence of words that co-occur more often than what would be expected by chance. We extract the collocation by analysing the word pairs extracted from 5-grams and score each word pair using the Dice score:

$$dice(w_a, w_b) = \frac{2 * f_{ab}}{f_a + f_b} \quad (1)$$

where f_{ab} is the number of times that the words w_a and w_b occur together and f_a and f_b are respectively the number of times that w_a and w_b occur in the corpus. Since the Dice score is independent of the corpus size, it is possible to build for each word and each time period a list of collocations by considering only the collocations occurring in a specific period of time. In order to consider only a restricted number of collocations, we take in account only the collocations with a Dice value above 0.0001. For each word and each time period we obtain a list of collocations with the associated Dice score. For example, a portion of the list of collocations for the word *pace (peace)* in the period 1980-1984 is reported as follows:

```
pace guerra 0.007223173
pace giustizia 0.0068931305
pace trattati 0.0067062946
pace trattative 0.006033537
```

Temporal Random Indexing (TRI). TRI (Jurgens and Stevens, 2009) is able to build a word space for each time period where each space is comparable to one another. In each space, a word is represented by a dense vector and it is possible to compute the cosine similarity between word vectors across time periods. In order to build comparable word spaces, TRI relies on the incremental property of the Random Indexing (Sahlgren, 2005). More details are provided in (Basile et al., 2014) and (Basile et al., 2016).

Temporal Word Analogies (TWA). This approach is able to build diachronic word embeddings starting from independent embedding spaces for each time period. The output of this process is a common vector space where word embeddings are used for computing temporal word analogies: word w_1 at time t_i is like word w_2 at time t_j. We build the independent embedding spaces by using the C implementation of word2vec with default parameters (Mikolov et al., 2013). More details about this approach are reported in (Szymanski, 2017).

Procrustes (HIST). This approach aligns the learned low-dimensional embeddings by preserving cosine similarities across time periods. More details are available in (Hamilton et al., 2016). We apply the alignment to the same embeddings created for TWA.

All approaches are built using the same vocabulary and the same context words generated starting from the 5-grams as previously explained.

3.2 Building the time series

In order to track how the semantics of a word changes over time we need to build a time series.

A time series is a sequence of values, one for each time period, that indicates the semantic shift of that word in the specific period. In our evaluation, we split the interval [1900-2012] in time periods of five years each.

The time series are computed in different ways according to the strategy used for representing the words. In particular, the values of each time series $\Gamma(w_i)$ associated to the word w_i is computed as follow:

- Collocation: given two lists of collocations related to two different periods, we compute the cosine similarity between the two lists by considering a list as a Bag-of-Collocations (BoC). In this case each point k of the series $\Gamma(w_i)$ is the cosine similarity between the BoC at time T_{k-1} and the BoC at time T_k;

- TRI: we use two strategies, (*point-wise* and *cumulative*), as proposed in (Basile et al., 2016). The point-wise approach captures how the word vector changes between two time periods, while the cumulative analyses captures how the word vector changes with respect to all the previous periods. In the point-wise approach, each point k of $\Gamma(w_i)$ is the cosine similarity between the word vector at time T_{k-1} and the word vector at time T_k, while for the cumulative approach the point k is computed as the cosine similarity between the average word vectors of all the previous time periods $T_0, T_1, \ldots, T_{k-1}$ and the word vector at time T_k;

- TWA: we exploit the word analogies across time and the common vector space for capturing how a word embedding changes across two time periods as reported in (Szymanski, 2017);

- HIST: time series are built by using the pairwise similarity as explained in (Hamilton et al., 2016).

We obtain seven time series as reported in Tables 1 and 2. In particular: BoC is build on temporal collocations; TRI_{point} and TRI_{cum} are based on TRI by using respectively point-wise and cumulative approach; TWA_{int} and TWA_{uni} are built using TWA on words that are common (intersection) to all the periods (TWA_{int}) and on the union of words (TWA_{uni}). The same procedure is used for $HITS$ obtaining the two time series $HIST_{int}$ and $HIST_{uni}$.

For finding significant change points in a time series, we adopt the strategy proposed in (Kulkarni et al., 2015) based on the Mean Shift Model (Taylor, 2000).

3.3 Metrics

We compute the performance of each approach by using Precision, Recall and F-measure. However, assessing the correctness of the change points generated by each system is a not easy task. A change point is defined as a pair *(lemma, year)*. In order to adopt a soft match, when we compare the change points provided by a system with respect to the change points reported in the gold standard, we take into account the absolute value of the difference between the year predicted by the system and the year provided in the gold standard.

As a first evaluation (exact match), we impose the difference between the detected year and the gold standard to be less or equal than five, which is the time period span of our corpus. As a second evaluation (soft match), we impose only that the predicted year is greater or equal than the change point in the gold standard. This is a common methodology adopted in previous work.

For a fairer evaluation, we perform the following steps:

- We remove from the gold standard all the change points that are outside of the period under analysis ([1900-2012]);

- We remove from the gold standard all the words that are not represented in the model under evaluation. This operation is necessary because (1) the previous filtering step can exclude some words;(2) there are words that do not appear in the original corpus.

Since the gold standard contains lemmas and not words, we perform a lemmatization of each output by using Morph-it! (Zanchetta and Baroni, 2005).

3.4 Results

Results of Precision (P), Recall (R) and F-measure (F) are reported in Table 1. We can observe that generally we obtain a low F-measure. This is due to a large number of false positive change points detected by each system.

Γ	exact match			soft match		
	P	R	F	P	R	F
BoC	.0034	.0084	.0049	.0274	.0670	.0389
TRI_{point}	.0056	.0394	.0098	.0248	.1750	.0434
TRI_{cum}	.0058	.0387	**.0101**	.0251	.1672	**.0436**
TWA_{int}	.0034	.0009	.0015	.0165	.0046	.0072
TWA_{uni}	.0052	.0060	.0056	.0373	.0435	.0402
$HIST_{int}$	.0024	.0048	.0032	.0111	.02211	.0148
$HIST_{uni}$	.0022	.0066	.0033	.0118	.0356	.0177

Table 1: Results of the evaluation.

Γ	exact match			soft match		
	P	R	F	P	R	F
BoC	.0361	.1243	.0560	.2881	.9930	.4466
TRI_{point}	.0581	.2244	.0923	.2581	.9973	.4100
TRI_{cum}	.0610	.2308	**.0959**	.2617	.9979	.4146
TWA_{int}	.0402	.2000	.0670	.1960	.9750	.3264
TWA_{uni}	.0526	.1367	.0759	.3794	.9866	**.5480**
$HIST_{int}$	.0344	.2147	.0593	.1569	.9791	.2704
$HIST_{uni}$	.0314	.1842	.0536	.1675	.9836	.2863

Table 2: Results of the evaluation obtained by considering only common lemmas between the gold standard and the system output.

The best approach in both evaluations is TRI_{cum}. Considering the *exact match* evaluation, the difference in performance is remarkable since generally TRI has a high recall. In the *soft match* evaluation, TWA_{uni} obtains the best precision, while the simple BoC method is able to achieve good results compared with more complex approaches such as TWA_{int} and $HIST$.

The results of the evaluation prove that the task of semantic change detection is very challenging; in particular, the large number of false positive drastically affects the performance.

Further analyses are necessary to understand which component affects the performance. In this preliminary evaluation, we adopt a unique approach for detecting the semantic shift. An extended benchmark is necessary for evaluating several approaches for detecting semantic change points.

The systems are built on a vocabulary that is larger than both the original dictionary and the gold standard. For that reason, we provide an additional evaluation in which we perform an ideal analysis by evaluating only lemmas that are common to the gold standard and the system output. The goal of this analysis is to measure the ability of correctly identifying change points for those lemmas that are represented in both the gold standard and the system. Results of this further evaluation are provided in Table 2

For the *exact match* evaluation, TRI_{cum} obtains the best F-measure as in the first evaluation, while TWA_{uni} achieves a very good performance in the *soft match* evaluation.

The plot in Figure 2 reports how the F-measure increases according to the time span that we adopt in the soft match. In particular, the X-axis reports the maximum absolute difference between the year in the gold standard and the year predicted by the system. We can observe that under 20 years TRI provide better performance than TWA, and after 60 years all the approaches reach a stable F-measure value.

4 Conclusion and Future Work

In this paper, we provide details about the construction of a dataset for the evaluation of semantic change point detection algorithms. In particular, our dataset focused on the Italian language and it is built by adopting a web-scraping strategy. We provide a usage example of our dataset by evaluating several approaches for the representation of words over time. The results prove that the task of detecting semantic shift is challenging due to a large

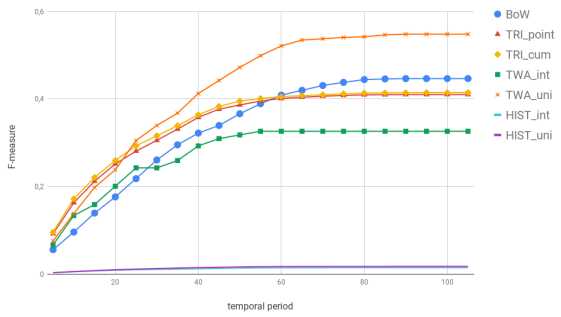

Figure 2: The plot shows how the F-measure increases according to the time span used in the soft match.

number of detected false positive. As future work, we plan to investigate further methods for building time series and detecting semantic shifts in order to improve the overall performance. Moreover, we plan to fix some issues of our extraction process in order to improve the quality of the dataset itself.

Acknowledgements

This work was supported by the ADAPT Centre for Digital Content Technology, funded under the Science Foundation Ireland (SFI) Research Centres Programme (Grant SFI 13/RC/2106) and is co-funded under the European Regional Development Fund and by the European Unions Horizon 2020 (EU2020) research and innovation programme under the Marie Skodowska-Curie grant agreement No.: EU2020 713567. The computational work has been executed on the IT resources made available by two projects, ReCaS and PRISMA, funded by MIUR under the program "PON R&C 2007-2013".

References

Pierpaolo Basile, Annalina Caputo, and Giovanni Semeraro. 2014. Analysing word meaning over time by exploiting temporal random indexing. In *First Italian Conference on Computational Linguistics CLiC-it*.

Pierpaolo Basile, Annalina Caputo, Roberta Luisi, and Giovanni Semeraro. 2016. Diachronic analysis of the italian language exploiting google ngram. *CLiC it*, page 56.

Filip Ginter and Jenna Kanerva. 2014. Fast training of word2vec representations using n-gram corpora.

William L Hamilton, Jure Leskovec, and Dan Jurafsky. 2016. Diachronic word embeddings reveal statistical laws of semantic change. *arXiv preprint arXiv:1605.09096*.

David Jurgens and Keith Stevens. 2009. Event detection in blogs using temporal random indexing. In *Proceedings of the Workshop on Events in Emerging Text Types*, pages 9–16. Association for Computational Linguistics.

Vivek Kulkarni, Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2015. Statistically significant detection of linguistic change. In *Proceedings of the 24th International Conference on World Wide Web*, pages 625–635. International World Wide Web Conferences Steering Committee.

Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K Gray, Joseph P Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, et al. 2011. Quantitative analysis of culture using millions of digitized books. *science*, 331(6014):176–182.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Magnus Sahlgren. 2005. An introduction to random indexing.

Terrence Szymanski. 2017. Temporal word analogies: Identifying lexical replacement with diachronic word embeddings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 448–453, Vancouver, Canada, July. Association for Computational Linguistics.

Nina Tahmasebi, Lars Borin, and Adam Jatowt. 2019. Survey of computational approaches to lexical semantic change. *arXiv:1811.06278v2*.

Wayne A Taylor. 2000. Change-point analysis: a powerful new tool for detecting changes.

Eros Zanchetta and Marco Baroni. 2005. Morph-it! a free corpus-based morphological resource for the italian language. In *Proceedings of corpus linguistics*.