Finding Individual Word Sense Changes and their Delay in Appearance

Nina Tahmasebi
Språkbanken,
University of Gothenburg, Sweden
nina.tahmasebi@gu.se

Thomas Risse
University Library J.C. Senckenberg,
Germany
t.risse@ub.uni-frankfurt.de

Abstract

We present a method for detecting word sense changes by utilizing automatically induced word senses. Our method works on the level of individual senses and allows a word to have e.g. one stable sense and then add a novel sense that later experiences change. Senses are grouped based on polysemy to find linguistic concepts and we can find broadening and narrowing as well as novel (polysemous and homonymic) senses. We evaluate on a testset, present recall and estimates of the time between expected and found change.

1 Introduction

When interpreting the content of historical documents, knowledge of changed word senses play an important role. Without knowing that the meaning of a word has changed (*word sense change*) we might falsely place a more current meaning on the word and thus interpret the text wrongly.

Recent work on detecting word sense change utilize word embeddings and have several drawbacks: (i) they look at all senses of a word at once and thus only track changes in a word's dominant sense, (ii) they can find *when* a word changes but not *what* has changed; and (iii) they cannot separate stable senses from changing senses for a word, e.g. the *stone* sense of *rock* stayed stable while a *music* sense was added and later changed.

In this paper, we propose a method that utilizes automatically extracted word senses by means of word sense induction, to find sense changes given a text collection. We test the hypothesis that automatically induced word senses and the temporal comparison of these has the potential to capture changes in all senses of a word separately. We apply unsupervised methods and show the potential of our method on a set of words that have experienced change in the past centuries. We perform word sense change detection per sense rather than considering a word and all its senses as one. We measure the time between an expected change in word sense and the corresponding found change to investigate not only *if* but *when* changes can be found and with which time delay.

We consider continuous data from two centuries, which leads to a high complexity; If all senses of a word can relate between adjacent time periods, the relation graph would result in a combinatorial explosion. Therefore, we reduce complexity by first detecting coherent senses over time and then comparing these. Nonetheless, the complexity is high with many relations to evaluate. Because we lack automatic evaluation methods and common testsets, we present a proof-of-concept of our method.

The contributions of this paper are as follows:

- Methodology for tracking word sense changes for individual senses.
- Analysis of time delay for detected changes with respect to ground truth.
- Testset for word sense change detection, the WSC dataset (Tahmasebi and Risse, 2017).

2 State of the Art

The first methods for automatic sense change detection were based on context vectors; they investigated semantic density (Sagi et al., 2009) and utilized mutual information scores (Gulordava and Baroni, 2011) to identify semantic change. Both methods detect signals of change but neither aligns senses over time or determines what changed.

Topic-based models (where topics are interpreted as senses) have been used to detect novel senses in one collection compared to another by identifying new topics in the later corpus ((Lau et al., 2012; Cook et al., 2014)), or to cluster top-

Proceedings of Recent Advances in Natural Language Processing, pages 741–749,
Varna, Bulgaria, Sep 4–6 2017.
https://doi.org/10.26615/978-954-452-049-6_095

Table 1: Comparison of evaluation in previous work.

	words		Time		Change
Reference	**# pos.**	**# neg.**	**# points**	**# span**	**# types**
Lau et al. (2012)	5	5	2	43	1
Cook et al. (2014)	20	204	2	43/17	1
Sagi et al. (2009)	4	0	4	560	2
Gulordava and Baroni (2011)	x	100-x[1]	2	30	1
Mitra et al. (2014)	69	0	8	488[2]	4
Kulkarni et al. (2015)	20[3]	0	21/12/24	105/15/2	1
Hamilton et al. (2016)	28[4]	0	20	200/190	1
This paper	35	26	222	222	4

[1] A random subset of 100 words were chosen. pos./neg. ratio unclear.
[2] The first portion of the data consists of 1520-1908 and contains roughly the same amount of data as the last portion spanning 2006-2008.
[3] An additional 20 words/method evaluated, pos./neg. ratio unclear.
[4] Same as [3] but 10 words/method

ics over time (Wijaya and Yeniterzi, 2011). A dynamic topic model that builds topics with respect to information from the previous time point is proposed by Frermann and Lapata (2016) and again sense novelty is evaluated. With the exception of Wijaya et al. that partition topics, no alignment is made between topics to allow following diachronic progression of an individual sense.

Graph-based models (Tahmasebi, 2013; Mitra et al., 2014, 2015) aim at revealing complex relations between a word's senses by (a) modeling senses per se using WSI; and (b) aligning senses over time. The models allow identification of individual senses in different time periods and Tahmasebi also groups senses into related concepts.

The largest body of work is done using word embeddings of different kind in the last years (Kim et al., 2014; Basile et al., 2016; Zhang et al., 2016). Embeddings are trained on different time-sliced corpora and compared over time. Kulkarni et al. (2015) project words onto their frequency, POS and word embeddings and propose a model for detecting statistically significant changes between time periods on those projections. Hamilton et al. (2016) investigate both similarity between a priori known pairs of words, and between a word's own vectors over time to detect change. Kulkarni et al. (2015); Basile et al. (2016); Hamilton et al. (2016) all propose different methods for projecting vectors from different time periods onto the same space to allow comparison.

Word embeddings do not allow us to recover the senses that have changed and therefore, no way of detecting *what* changed. Most methods use similar words to the changing word as a method to illustrate what happens. However, the most similar words will only represent the dominant sense and not reflect changes among the other senses or capture stable parts of a word.

Table 1 shows a summary over the evaluation performed by some of the methods described. We present the number of positive and negative examples of change, the number of time points and the total timespan as well as the number of considered change types. Currently no standard datasets or evaluation metrics are available for comparison across methods or datasets.

3 Modeling Word Sense Change

As a basis for our analysis we consider automatically induced word sense clusters. Each cluster represents a distinct time period and consists of a set of nouns and noun phrases of length two. These clusters are approximations of word senses and to some extent capture also contexts. Throughout the paper we use **word senses** and **clusters** interchangeably and refer to these automatically derived approximations. A **concept** consists of senses that are related (i.e., polysemous) following Cooper (2005). To model word sense change, we should allow each sense to change individually; worst case, this results in a graph where, for a maximum number of senses S in each time period $t \in T$, we have in the order of $S^{|T|}$ edges representing sense similarity. Even for a small number of time periods, this graph becomes infeasible to investigate and evaluate. Therefore, we reduce this complexity by first considering coherent senses over time (units) and then following the units over time. Units that are related are placed in a *path*. A unit can contain an arbitrary number of clusters so in order to get a good representation, we create a *unit representative*.

We define a **unit** $u_i(w) = u_i$ as an ordered sequence of clusters $\{c_1^{t_i}, c_2^{t_j}, \ldots, c_n^{t_k}\}$ such that each cluster contains the word w and comes from a distinct time period t_j where $t_i \leq t_j \leq t_k$. We allow time gaps between the clusters, i.e., $t_k \geq t_j + 1$, in order to capture senses that have lost in popularity or are underrepresented for a period of

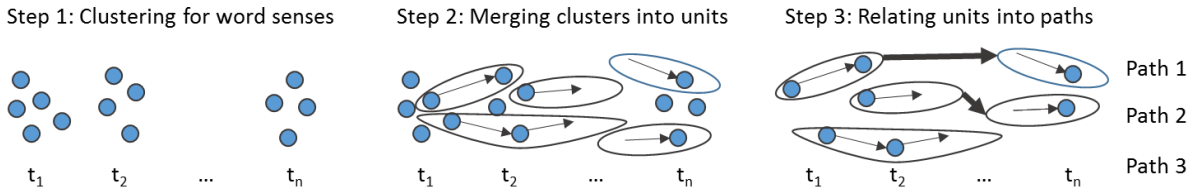

Figure 1: Process for word sense change detection.

time. The unit representative, r_{u_i}, contains a set of words that can be used to represent all participating clusters. A **single unit** is a unit with one cluster where the cluster words are the unit representative. A unit set $U_{t_i}(w)$ consists of all units for w that start at t_i. We measure **similarity between units** as similarity between unit representatives.

A **path** is an ordered sequence of units $\{u_i, u_j, \ldots, u_k\}$ such that units have a similarity above α. A path represents polysemous senses and all their changes for a word. Different paths represent homonymic concepts.

4 Methodology

Our method consists of three steps (Figure 1). We begin with an example and then provide details.

We choose an example with three time points t_1, t_2, t_3 and unit sets $U_{t_1}(w) = \{u_1\}$, $U_{t_2}(w) = \{u_2, u_3\}$ and $U_{t_3}(w) = \{u_4, u_5\}$ for the target word *tape*. Each unit represents a cluster chosen from the word sense change (WSC) dataset.
$u_1 = \{$*stereo, cassette, tape, radio, record*$\}$,
$u_2 = \{$*pin, thread, tape, silk, chair, cotton*$\}$,
$u_3 = \{$*tape, radio, record, cassette*$\}$,
$u_4 = \{$*tape, sparkplug cable, wire, clip*$\}$,
$u_5 = \{$*television, record, tape, video, book, film, magazine, video industry*$\}$.

In the first step, similarity between pairs (u_1,u_2) and (u_1,u_3) is measured. Pairs are ranked according to similarity and the pair with the highest similarity is merged. In this case, u_1 and u_3 are merged into $u' = \{u_1, u_3\}$ because u_3 is a subset of u_1. The unit representative consists of the words $\{$*cassette, tape, record*$\}$. The pair (u_1,u_2) is removed because u_1 is merged with one unit from $U_{t_2}(w)$.

The resulting unit set is $U_{[t_1,t_2]}(w) = \{\{u_1, u_3\} = u', u_2\}$. At time t_3, unit u_4 and u_5 are compared to the two units in $U_{[t_1,t_2]}(w)$. u_5 is merged with u' resulting in $u'' = \{u_1, u_3, u_5\}$. u_4 remains a single unit and is placed in $U_{[t_1,t_3]}(w)$ without being merged. When we merge two units, we add up all their clusters and build a new representative. When unit u_5 is merged with $u' = \{u_1, u_3\}$ we consider this to be a broadening because the single unit u_5 has a broader sense than the merged unit u'. The resulting unit set consists of $U_{[t_1,t_3]}(w)$=$\{\{u_1, u_3, u_5\}$=$u'', u_2, u_4\}$.

As a final step, to create paths, we measure similarity between the pairs (u'',u_2) and (u'',u_4). In this example, no units are related into paths (i.e., they all form their own path) which tells us that there are three different concepts for *tape*, one regarding *sewing tape*, one regarding *scotch tape* and one regarding *musical tape* which later includes also the *video tape*, matching well the main senses of tape but also capturing *sewing tape*, a sense less common today (OED, 2000).

4.1 Deriving Word Sense Clusters

We find word senses using an unsupervised word sense induction algorithm called *curvature clustering* (Dorow et al., 2005). These clusters have been evaluated by Tahmasebi et al. (2013) and were shown to have 85% precision using the Pantel and Lin (2002) evaluation method. To the best of our knowledge, the curvature clustering method is the only induction method that has been evaluated on historical texts and therefore, is used as a starting point for our word sense change detection. However, our method can make use of senses derived by any induction algorithm, or the combination of several, where the output is a set of words.

Measuring Unit Similarity A central part of word sense tracking is to measure similarity between the induced senses (by means of unit similarity). We consider equality between words $w_i \in r_{u_i}$ and $w_j \in r_{u_j}$ in two ways; 1. full match and 2. partial match. If there is no full match, we split all words into their (space separated) parts $w_i = w_1\ w_2$ and a word w_i is accepted as a partial match to w_j only if any w_1 or w_2 is a suffix or prefix in w_j, e.g. *motor car* and *motorcar* as well as *monitor* and *color monitor* but not *rave* and *gravel*.

Our similarity measure, lin (Lin, 1998; Pantel and Lin, 2002), is semantic and based on closeness of WordNet synsets (Miller, 1995). We measure similarity between clusters by considering them as single units and measuring unit similarity.

$$sim(u_i, u_j) = \frac{\sum_{u \in r_{u_i}} \max_{v \in r_{u_j}} lin(u, v)}{\min(|r_{u_i}|, |r_{u_j}|)} \quad (1)$$

4.2 Creating Units

We use the unit similarity to determine which clusters represent coherent senses and merge these into units. In the merging step, we calculate the similarity between all pairs of units (u, v) such that $u \in U_{[t_1,t_k]}(w)$ and $v \in U_{t_{k+1}}(w)$. In the first step we set $k = 1$ and simply perform merging by looking at clusters (i.e., single units) from t_1 and t_2. In the subsequent steps, we merge single units from time t_{k+1} with all units up to time k.

For each unit v, we normalize similarity s.t. the sum of the similarities between v and all units u amounts to 1 and keep only pairs with similarity above a threshold α. We then merge the pair (v, u) with the highest similarity and remove all pairs $(u, *)$ and $(*, v)$. If the similarity of v and several units u is the same, we assume that there are polysemous concepts. By uniquely assigning and merging v with one unit u we reduce the complexity of the method. It remains future work to investigate how much information is lost by not allowing a more complex graph structure.

Capturing the Core of Units Once we determined which clusters that should form a unit, we need a representation of the participating clusters, i.e., we need to chose the words for the unit representative. For this, we make use of *local clustering coefficient* (lcc) (Watts and Strogatz, 1998).We also measure participation rate $part$ as the amount of participating clusters where the word is present. To allow a word in the unit representative, the word must be highly central in its cluster (high lcc) or participate in many clusters (high $part$). Each time a new cluster is added, we update the unit representative, and thus allow a slow shift of the unit representatives capturing broadening, narrowing and evolved senses (e.g. adding *video* to the *music tape* sense).

4.3 Creating Paths

Once all clusters and units have been merged, the final set $U_{[t_{start},t_{end}]}(w)$ consists of units that represent individual senses over time. To find concepts, we seek to group units s.t. the units are related, i.e., have a similarity above α, following an idea by Mei and Zhai (2005). We compare each unit $u \in U_{[t_{start},t_{end}]}(w)$ to all other units that start at the same or later time point. We allow time gaps between units to capture relations between underrepresented senses. All units that are related are placed in a $path$.

Table 2: Extract of units for *aeroplane*. Units only display some of the internal clusters and words.

Year	Cluster words
	Unit $\boldsymbol{u}_1$: 1908-1930 (defining the construction)
1908	airship, aeroplane, balloon, aeroplane construction
1930	aeroplane, automobile, airship, engine work, liner
	Unit $\boldsymbol{u}_2$: 1917-1943 (as a weapon of war)
1917	piping, gun, aeroplane, shafting, tank, infantry
1933	armoured car, aeroplane, tank
1943	tank, aeroplane, ship, gun, ammunition
	Unit $\boldsymbol{u}_3$: 1914-1941 (as a means of transportation)
1914	plane, aeroplane, motor bicycle, motor lorry, car
1920	train, lorry, car, aeroplane
1930	motorcycle, lorry, motorcar, aeroplane
1941	tank, machineguns, gun, lorry, motorcycle, aeroplane
	Unit $\boldsymbol{u}_4$: 1916-1974 (unit with all senses)
1916	aeroplane, bird, ship
1930	train, aeroplane, ship
1941	gun, artillery, machineguns, aeroplane, ship, tank
1974	car, train, aeroplane, motor car

Table 2 shows units for *aeroplane*. The three first units represent the individual senses and the last unit contains the changes all in one unit. We find two paths $u_1 \rightarrow u_2 \rightarrow u_4$, and $u_3 \rightarrow u_4$.

5 Experiments

The aim of our experiments is to find the quality and degree (i.e., recall) to which word sense change can be found using our proposed methodology against the main changes according to a set of knowledge sources. We use *The Times Archive*, a large sample of modern English spanning 1785 – 1985 and append the *New York Times Annotated Corpus*, a modern collection spanning 1987 – 2007, giving us a total of 222 years.

5.1 Testset

As a testset, we manually chose a set of 23 words that we know have experienced word sense change during the past centuries. The main changes for each word were found using Wikipedia, dictionary.com and the Oxford English Dictionary, and the automatically found changes were compared

against the manually found counterpart. For comparison purposes we also chose a set of 11 words that have experienced minimal change during the period, i.e., **stable words**. The full testset can be found in (Tahmasebi and Risse, 2017).

We categorize changes into several classes:

- **evolved sense** A sense that changes by means of broadening or narrowing. Should be found within one unit, e.g. *mail* as *electronic mail*;
- **novel related sense** (polysemy) A sense that is related to at least one existing sense. Should be found as a new unit in an existing path, e.g. *record* as a *musical record* related to *official record*, to constitute a concept;
- **novel unrelated sense** (homonomy) A sense that is unrelated to any existing senses or corresponds to a new word (neologism). Should be found in a separate path, e.g. *Internet* or *rock as music* different from *rock as stone*;
- **existing sense** A sense that appears before the start of our dataset and is stable during the entire period, e.g. the *stone* sense of *rock*. This class is used for words without any change events (existing – stable) and words that later experience change (existing – evo).

The resulting testset consists of 16 evolved senses, 9 novel related and 10 novel unrelated senses, 15 existing senses for changing and 11 existing senses for stable words. To sum up, we have **35 change events** and **26 non-change events**.

5.2 Setup

We cluster using a minimum clustering coefficient of 0.3. A word w from any participating cluster is placed in the unit representative r_u if; (1) $lcc \geq 0.7$; (2) $part \geq 60\%$; or (3) if $lcc \geq 0.4$ and $part \geq 50\%$ hold. To filter out noise, we remove all single units once the paths are created.

We use the WordNet Similarity for Java implementation (WS4J, 2014) for the lin measure. We set $\alpha = 0.1$ to be as inclusive as possible while keeping the number of possible pairs down.

5.3 Evaluation

For each experiment, we firstly measure **recall** and discuss false positives; and secondly the **average time delay** as the difference in time between the *expected*, according to our ground truth, and the *found* events. Finally, we measure the **average path length** to see if we can differentiate between stable and changing words.

Recall is straightforward and measures the portion of expected change present in our paths, according to our ground truth. The expected time of change is more complex; true time of change is the first time that a word is used in a collection with the correct corresponding sense. We do not know this and therefore we approximate it using two different time points.

The first expected time point is the *time of definition* or *time of invention* of a word w, $t_{DI}(w)$, in a given dictionary or knowledge resource. However, that an invention has been made does not necessarily correspond to newspapers reporting on it frequently. E.g. the *computer* was invented in its modern form in the 1940s, but was not mentioned in newspapers often in the early 40's, most likely due to WWII. Therefore, as a second expected time point, we consider the *first cluster evidence*, $t_{CE}(w)$, indicating the first time the word appears in a cluster. This represents the first possible time point for tracking, given the curvature clustering algorithm for extracting word sense clusters. The true expected time point lies in the interval $[t_{DI}, t_{CE}]$. Finally, we have the time point of the detected change event, $t_{found}(w)$.

The time delay is $T_{DI}(w) = t_{found}(w) - t_{DI}(w)$ and $T_{CE}(w) = t_{found}(w) - t_{CE}(w)$. The average time delay is summed over all words, $AT_{DI} = \frac{\sum_{\forall w} T_{DI}(w)}{|w|}$ and $AT_{CE} = \frac{\sum_{\forall w} T_{CE}(w)}{|w|}$.

Experimental setup We split our experiments into two parts; In the first experiment, *best case experiment*, we investigate how much can be detected in the units. We do not make any distinction between different change events and we view this as an upper bound on our recall and a lower bound on avg. time delay. This experiment aims to answer the question; how suitable are the units for capturing the expected word senses and their changes? Implicitly, we capture the potential of the induced word senses as a basis for word sense change detection. Are the senses present among the induced senses and can the change events be found among the units?

The second experiment is to evaluate units and paths for capturing and differentiating between change events. We call this the *all classes experiment* and consider each class (existing, novel unrelated, novel related and evolved) separately. This is the full evaluation of our method and shows how well the classes can be differentiated. Thus we require each change type to appear in the correct

Table 3: Recall and time delay for all words in the testset, where *BC* is the best case and *All* is the all class experiments. The *value* in *bold* represents delay time from first cluster evidence AT_{CE} and the second represents time of definition AT_{DI}.

	Recall		Avg. time delay	
	BC	All	BC	All
Evolved sense	1.00	1.00	**6.2** - 11.0	**16.1** - 20.9
New related sense	0.89	0.11	**5.8** - 27.8	**26.0** - 36.0
New unrelated sense	0.80	0.80	**1.6** - 19.8	**1.9** - 20.0
Existing sense – evo	1.00	1.00	**11.7**- 59.0	**11.7**- 59.0
Existing – stable	1.00	1.00	**2.7** -20.5	**2.7** -20.5
Average excl. stable	0.94	0.80	**7.1** - 30.7	**11.8** - 35.4
Total average	0.95	0.84	**6.3** - 28.7	**9.9** - 32.2

form (see Sec. 5.1).

6 Experimental Results

6.1 Recall

Table 3 shows the recall for the best case (*BC*) and all classes (*All*). For the **evolved sense** class (broadening and narrowing within an existing unit), we have full recall and found all 16 events within units. For **new unrelated senses** (new units in their own path) we correctly found 8 out of 10. The only senses that are not found are the first senses for *Internet* and *computer*, most likely because of few mentions in the dataset. The **existing senses** are all found, regardless of if they are senses attached to a stable word or to a word that will later gain or change meaning.

The **new related sense** (new unit related to an existing path) is the hardest class to find. The units contain evidence for 89% of all changes, however, they cannot be found related to other units. By looking at examples from this class, it is obvious that the linguistic definition is very hard to detect automatically. E.g. the word *memory* in a digital sense is related to *human memory*, but rarely used in similar context and *train* as a *mechanical train* with a locomotive differs largely from a *train of people* (e.g. funeral train). Therefore, we cannot place them in the correct path and they are placed in their own path.

To sum up, our recall is 95% for all changes and stable senses in our units. In the correct form within the paths, we have a recall of 84%, only missing out on new related senses where the linguistic definition does not match the usage.

False positives Providing precision requires a definition of precision in the case of word sense change detection using paths. When do we achieve full precision? In a unit with 70-80 cluster or a path with hundreds of units, evaluation becomes extremely complex. Therefore, instead of precision, we analyze false positives by looking at the average number of change events per word and leave the definition of precision for future work.

On average, there are 3 paths/word and 5.3 units/path for change words and 13.3 for stable words. Among the changing words, we have an average of 2.2 change events and thus we would expect around 2 false positives (5.3 units mean 4 change events on average out of which we expect 2 to be correct). Among the stable words, all change events and thus different units are per definition wrong, that means on average 13.3 false positives. However, there are some words that stand out, *horse*, *bank* and *music* are very common words and have, on average, 47.5, 21.4 and 24.9 units per path when we would expect only one. For these we observe very long spanning units with 206, 197 and 204 years for the longest unit. Excluding these three words, the average number of unit per path drops to 6.6 and represents 5 change events.

Though this is an approximation of the false positive rate, it does tell us that the number of elements to manually filter is limited and thus the results can be of great use for digital archive users and researchers in e.g. the digital humanities.

6.2 Average Time Delay

Table 3 shows the average time delays for our experiments. Bold values are delay times with respect to first cluster evidence, AT_{CE} and the second values time of definition AT_{DI}.

For the **evolved sense** class, the change events are found in our units 6.2 years after first appearing in a cluster and 11 years after being invented or defined in a dictionary. We consider the true time delay to be between 6.2 – 11 years. To appear in the correct form, i.e., inside an existing unit, the average time delay is 16.1 – 20.9 year. For the **new related senses** we have a time delay of 5.8 – 27.8 years for the words to appear in any unit. However, to appear in the correct form, the time delay is much higher (26.0 – 36.0 years) and gives evidence for the fact that this class is very hard to detect using context-based methods.

The **new unrelated senses** have the lowest time delay of all classes and take between 1.6 – 19.8 years to appear in any unit and only marginally

more, 1.9 – 20.0, to appear in the correct form.

Existing senses show an interesting behavior; the existing senses for words that later have a change event have significantly longer average time delays compared to existing senses of stable words, 11.7 compared to 2.7. One possible explanation is that words are less likely to change their meanings, if they are commonly used and hence we cannot find them in our dataset.

For all change events, we find an average delay of 7.1 – 30.7 years for any evidence to appear in a unit and 11.8 – 35.4 for our method to find the change in its correct form (see Sec. 5.1). Including existing senses, all delay times decrease slightly. We consider 7.1 years to appear in a unit a reasonable time delay given the 222 year time span. The delay of 4.7 years (between 7.1 to 11.8) for the change to appear in the correct form could be decreased by optimizing thresholds and merging strategies. To find the reason for the upper limit (30.7 and 35.4 years) we need to use linguists and historians with in depth knowledge of the datasets, time period and place of publication.

6.3 Average Path Length

To further investigate if the stable senses can be differentiated from changing ones, we measure the *average length of a path* as the difference between the earliest to last participating cluster. We find that stable words have statistically significantly longer paths (181 years) than words that change their meanings over time (114 years), clearly differentiating the classes with our method.

7 Discussions

Our units capture 94% of all expected changes. As comparison, a natural baseline is concordances; where we would expect an upper bound close to 100%. For certain words, concordances are enough and can e used to deduce a new sense. However, mostly, an induction mechanism is needed which will result in reduced recall. Therefore, we consider our recall good evidence for the choice of methods for creating clusters and units.

Our similarity measure relies on WordNet that suffers from having a low coverage of older texts.We re-ran our experiments using a modified Jaccard similarity and found small differences that were not statistically significant. We find that set similarity measures can therefore offer a viable option for resource poor languages.

Our method goes beyond those using distributional semantics that embed words to vectors and detect changes by comparing the vectors. These methods can find changes in the dominant sense of a word but cannot differentiate between senses or allow some senses to stay stable while others change. We believe that the future lies in a combined approach, using embeddings (possibly multi-sense embeddings (Trask et al., 2015; Li and Jurafsky, 2015; Pelevina et al., 2016)) and sense-differentiated techniques.

8 Conclusions and Future Work

In this paper, we presented a method for word sense change detection that relies on an existing induction algorithm and uses the induced word senses as a basis for finding coherent senses (units) and grouping units into polysemous concepts (paths). By tracking individual sense changes, we can differentiate a word's changing senses from its stable ones.

On average, 94% of the change events and 95% of all events, including stable senses, were found with a time delay of between 6.3 and 7.1 from the first cluster evidence. Our *all classes* experiment shows how well the different change events can be differentiated. For the evolved sense category, we have a 100% recall. The new unrelated (homonymic) senses yield 80% recall. Only the new related (polysemous) category perform badly; a high-level linguistic relation is not captured in the context of a word.

Our method detected change in the correct form 9.9 – 11.8 years after the first cluster evidence and is the first work to report such time analysis. Given the 222 year timespan, we consider this delay to be a good starting point for future work. Moving forth, we will use a combined approach, utilizing the potential of embeddings with sense-differentiated graph-based techniques.

Acknowledgments

This work has been funded in part by a framework grant *Towards a knowledge-based culturomics*; contract 2012-5738), funding to Swedish CLARIN (*Swe-Clarin*; contract 2013-2003), both awarded by the Swedish Research Council, and by the European Research Council under Alexandria (ERC 339233). We would like to thank Times Newspapers Limited for providing the archive of The Times for our research.

References

Pierpaolo Basile, Annalina Caputo, Roberta Luisi, and Giovanni Semeraro. 2016. Diachronic analysis of the italian language exploiting google ngram. In *Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016)*.

Paul Cook, Jey Han Lau, Diana McCarthy, and Timothy Baldwin. 2014. Novel word-sense identification. In *Proceedings of COLING 2014*. Dublin, Ireland, pages 1624–1635. http://www.aclweb.org/anthology/C14-1154.

Martin C. Cooper. 2005. A Mathematical Model of Historical Semantics and the Grouping of Word Meanings into Concepts. *Computational Linguistics* 32(2):227–248. https://doi.org/10.1162/0891201054223995.

Beate Dorow, Jean-pierre Eckmann, and Danilo Sergi. 2005. Using curvature and markov clustering in graphs for lexical acquisition and word sense discrimination. In *Proceedings of the Workshop MEANING-2005*.

Lea Frermann and Mirella Lapata. 2016. A bayesian model of diachronic meaning change. *TACL* 4:31–45.

Kristina Gulordava and Marco Baroni. 2011. A distributional similarity approach to the detection of semantic change in the Google Books Ngram corpus. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*. Association for Computational Linguistics, Stroudsburg, PA, USA, GEMS '11, pages 67–71. http://dl.acm.org/citation.cfm?id=2140490.2140498.

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2016. Diachronic word embeddings reveal statistical laws of semantic change. *CoRR* abs/1605.09096. http://arxiv.org/abs/1605.09096.

Yoon Kim, Yi-I Chiu, Kentaro Hanaki, Darshan Hegde, and Slav Petrov. 2014. Temporal analysis of language through neural language models. In *Workshop on Language Technologies and Computational Social Science*.

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*. ACM, pages 625–635.

Jey Han Lau, Paul Cook, Diana McCarthy, David Newman, and Timothy Baldwin. 2012. Word sense induction for novel sense detection. In *EACL 2012, 13th Conference of the European Chapter of the Association for Computational Linguistics*. pages 591–601. http://aclweb.org/anthology-new/E/E12/E12-1060.pdf.

Jiwei Li and Dan Jurafsky. 2015. Do multi-sense embeddings improve natural language understanding? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. ACL, pages 1722–1732.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2*. Association for Computational Linguistics, Stroudsburg, PA, USA, ACL '98, pages 768–774. https://doi.org/10.3115/980691.980696.

Qiaozhu Mei and ChengXiang Zhai. 2005. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM* 38(11):39–41. https://doi.org/10.1145/219717.219748.

Sunny Mitra, Ritwik Mitra, Suman Kalyan Maity, Martin Riedl, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2015. An automatic approach to identify word sense changes in text media across timescales. *Natural Language Engineering* 21(05):773–798.

Sunny Mitra, Ritwik Mitra, Martin Riedl, Chris Biemann, Animesh Mukherjee, and Pawan Goyal. 2014. That's sick dude!: Automatic identification of word sense change across different timescales. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 USA*. pages 1020–1029. http://aclweb.org/anthology/P/P14/P14-1096.pdf.

OED. 2000. The Oxford English Dictionary 2nd ed. 1989. OED Online. Oxford University Press. 4 Apr. 2000. `http://dictionary.oed.com`.

Patrick Pantel and Dekang Lin. 2002. Discovering word senses from text. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'02)*. ACM, Edmonton, Alberta, Canada, pages 613–619. https://doi.org/10.1145/775047.775138.

Maria Pelevina, Nikolay Arefyev, Chris Biemann, and Alexander Panchenko. 2016. Making sense of word embeddings. In *Proceedings of the 1st Workshop on Representation Learning for NLP*. pages 174–183.

Eyal Sagi, Stefan Kaufmann, and Brady Clark. 2009. Semantic density analysis: comparing word meaning across time and phonetic space. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*. Association for Computational Linguistics, Stroudsburg, PA, USA, GEMS '09, pages 104–111. http://dl.acm.org/citation.cfm?id=1705415.1705429.

Nina Tahmasebi, Kai Niklas, Gideon Zenz, and Thomas Risse. 2013. On the applicability of word sense discrimination on 201 years of modern english. *International Journal on Digital Libraries* 13(3-4):135–153. https://doi.org/10.1007/s00799-013-0105-8.

Nina Tahmasebi and Thomas Risse. 2017. Word Sense Change Test Set. https://doi.org/10.5281/zenodo.495572.

Nina N. Tahmasebi. 2013. *Models and Algorithms for Automatic Detection of Language Evolution*. Ph.D. thesis, Gottfried Wilhelm Leibniz Universitt Hannover. http://edok01.tib.uni-hannover.de/edoks/e01dh13/771705034.pdf.

Andrew Trask, Phil Michalak, and John Liu. 2015. sense2vec - A fast and accurate method for word sense disambiguation in neural word embeddings. *CoRR* abs/1511.06388.

Duncan J. Watts and Steven Strogatz. 1998. Collective dynamics of "small-world" networks. *Nature* 393:440–442.

Derry Tanti Wijaya and Reyyan Yeniterzi. 2011. Understanding semantic change of words over centuries. In *Proceedings of the 2011 international workshop on DETecting and Exploiting Cultural diversiTy on the social web*. ACM, New York, NY, USA, DETECT '11, pages 35–40. https://doi.org/10.1145/2064448.2064475.

WS4J. 2014. WordNet Similarity for Java. `https://code.google.com/p/ws4j/`. [Online; accessed 2014-09-23].

Yating Zhang, Adam Jatowt, and Katsumi Tanaka. 2016. Detecting evolution of concepts based on cause-effect relationships in online reviews. In *Proceedings of the 25th International Conference on World Wide Web*. ACM, pages 649–660.