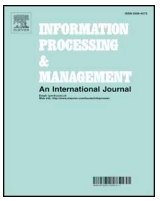
Information Processing and Management 57 (2020) 102173

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman



Network measures: A new paradigm towards reliable novel word sense detection

Abhik Jana*, Animesh Mukherjee, Pawan Goyal

Department of CSE, IIT Kharagpur, India

ARTICLE INFO

Keywords:
Novel sense detection
Distributional thesaurus network
Complex network measures

ABSTRACT

In this era of digitization, with the fast flow of information on the web, words are being used to denote newer meanings. Thus novel sense detection becomes a crucial and challenging task in order to build any natural language processing application which depends on the efficient semantic representation of words. With the recent availability of large amounts of digitized texts, automated analysis of language evolution has become possible. Given corpus from two different time periods, the main focus of our work is to detect the words evolved with a novel sense precisely. We pose this problem as a binary classification task to detect whether a new sense of a target word has emerged. This paper presents a unique proposal based on network features to improve the precision of this task of detecting emerged new sense of a target word. For a candidate word where a new sense has been detected by comparing the sense clusters induced at two different time periods, we further compare the network properties of the subgraphs induced from novel sense clusters across these two time periods. Using the mean fractional change in edge density, structural similarity and average path length as features in a Support Vector Machine (SVM) classifier, manual evaluation gives precision values of 0.86 and 0.74 for the task of new sense detection, when tested on 2 distinct time-point pairs, in comparison to the precision values in the range of 0.23-0.32, when the proposed scheme is not used. The outlined method can, therefore, be used as a new post-hoc step to improve the precision of novel word sense detection in a robust and reliable way where the underlying framework uses a graph structure. Another important observation is that even though our proposal is a post-hoc step, it can be used in isolation and that itself results in a very decent performance achieving a precision of 0.54-0.62. Finally, we also show that our method is able to detect well-known historical shifts in 80% cases.

1. Introduction

With the advancement of technology, a huge amount of information is floating around the Web which is expressed in natural language. Detecting the sense of a word is a primary step for natural language understanding. Researchers have tried to detect the sense of words from a given corpus in both supervised and unsupervised ways. One stream of work deals with the task of word sense induction (WSI), the goal of which is to automatically induce different senses of a given word, generally in the form of an unsupervised learning task with senses represented as clusters of words. The word sense disambiguation (WSD) task opens up another stream of literature, where a fixed sense inventory is assumed to exist, and the senses of a given word are disambiguated using the sense inventory as a reference. However, in both of these tasks, the assumption is that the number of senses that a word has is static.

* Corresponding author.
E-mail addresses: abhik.jana@iitkgp.ac.in (A. Jana), animeshm@cse.iitkgp.ac.in (A. Mukherjee), pawang@cse.iitkgp.ac.in (P. Goyal).

https://doi.org/10.1016/j.ipm.2019.102173
Received 5 April 2019; Received in revised form 17 November 2019; Accepted 18 November 2019
Available online 28 November 2019
0306-4573/

In addition, the senses do exist in the sense inventory to compare with for these tasks. They attempt to detect or induce one of these senses depending on the context.

Natural language, however, is dynamic and is constantly evolving as per the users' needs which leads to continuous changes of word meanings over time. For example, by late 20th century, the word 'virus' has come up with the 'technology' related sense whereas the word 'cool' has emerged with 'smart, calm personality' related sense. How to automatically detect such changes? Given sufficient time-stamped data, can one design efficient algorithms to detect known as well as unknown shifts in word meanings highly precisely? Such automation can directly benefit people such as librarians, historians or linguists who work with digitized texts from different time periods. The variation in the sense of a word could either be attributed to the emergence of a completely new sense of the word or change of usage of a well-established sense of the word. This semantic evolution of a word over time is of great interest to historical linguistics. Besides, lexicography is also expensive; compiling, editing and updating sense inventory entries frequently remains cumbersome and labor-intensive.

In general, detecting time-specific knowledge would make word meaning representations more accurate and hence, is a worthy problem to study for specialists such as etymologists, librarians, general public and NLP researchers. A well constructed semantic representation of a word is useful for many natural language processing or information retrieval systems like machine translation, semantic search, disambiguation, Q&A, etc. Taking into account the newer senses of a word can increase the relevance of the query result leading to a better semantic search. Similarly, a sense disambiguation engine informed with the newer senses of a word can increase the efficiency of disambiguation; it can recognize senses not available in sense inventory that would otherwise be wrongly assigned to any of the available senses from the inventory. Above all, a system having the ability to perceive the novel sense of a word can help in an automatic sense inventory update by taking into account the temporal scope of senses.

1.1. Recent advancements

Investigations of individual words or phrases were very limited due to the need for huge human involvement until now. Recently, with the arrival of large-scale collections of historic texts and online libraries such as Google books, a new paradigm has been added to this research area, whereby the prime interest is in identifying the temporal scope of a sense (Gulordava & Baroni, 2011; Jatowt & Duh, 2014; Lau, Cook, McCarthy, Gella, & Baldwin, 2014; Tahmasebi, Risse, & Dietze, 2011) which, in turn, can give further insights to the phenomenon of language evolution. Some recent attempts (Eger & Mehler, 2016; Frermann & Lapata, 2016; Hamilton, Leskovec, & Jurafsky, 2016a; 2016b; Kulkarni, Al-Rfou, Perozzi, & Skiena, 2015) have been made to model the dynamics of language in terms of word senses.

One of the studies in this area has been presented by Mitra et al. (2014) where the authors show that at earlier times, the sense of the word 'sick' was mostly associated to some form of illness; however, over the years, a new sense associating the same word to something that is 'cool' or 'crazy' has emerged. Their study is based on a unique network representation of the corpus called a distributional thesauri (DT) network built using Google books syntactic n-grams. They have used unsupervised clustering techniques to induce a sense of a word and then compared the induced senses of two time periods to get the new sense for a particular target word.

1.2. Limitations of the recent approaches

While Mitra et al. (2014) reported a precision close to 0.6 over a random sample of 49 words, we take another random sample of 100 words separately and repeat manual evaluation. When we extract the novel senses by comparing the DTs from 1909–1953 and 2002–2005, the precision obtained for these 100 words is as low as 0.32. Similarly, if we extract the novel senses comparing the DTs of 1909–1953 with 2006–2008, the precision stands at 0.23. We then explore another unsupervised approach presented in Lau et al. (2014) over the same Google books corpus,[1] apply topic modeling for sense induction and directly adapt their similarity measure to get the new senses. Using a set intersecting with the 100 random samples for Mitra et al. (2014), we obtain the precision values of 0.21 and 0.28, respectively. Clearly, none of the precision values are good enough for reliable novel sense detection. Thus, the primary motivation of this work is to devise an approach that is able to boost the precision values to an acceptable range. However, the idea is not to build the entire framework from scratch but to carefully re-engineer the algorithm presented by Mitra et al. (2014). Precisely, we show how we can take extreme advantage of the differences in certain properties of the networks that the authors had originally built to detect sense changes. Note that these properties were completely overlooked in the original approach and our main contribution is to unfold their tremendous benefit in improving the precision of novel sense detection.

It is well known that network science has proved to be very effective in addressing problems related to various complex phenomena including the structure and dynamics of the human brain, the functions of genetic pathways, the social behavior of humans in the online and offline world and many more. Some recent work shows that complex network concepts are being applied to understand human languages as well (Antiqueira, Nunes, Oliveira Jr, & Costa, 2007; Ferrer i Cancho, Capocci, & Caldarelli, 2007). The ability to access embedded knowledge makes complex networks extremely promising for natural language processing which normally requires deep knowledge representation. Many works exist where network properties are applied to natural language processing tasks, which lead to elegant solutions to the problem. Examples include word co-occurrence network (Ferrer i Cancho & Solé, 2001), word association network (Bonneau, Just, & Matthews, 2010), syntactic dependency network (Ferrer i Cancho, 2004),

[1] http://commondatastorage.googleapis.com/books/syntactic-ngrams/index.html, we use 'triarcs' dataset from 'English All'

etc. Some other applications of complex networks in NLP include ways to evaluate machine-generated summaries (Pardo, Antiqueira, Nunes, Oliveira Jr, & da Fontoura Costa, 2006), detection of ambiguity in a text (Dorow et al., 2004), etc. These works constitute our prime motivation to apply network science methods to enhance the precision of novel word sense detection.

1.3. Our proposal and the encouraging results

We propose a supervised method based on the network features to reduce the number of false positives and thereby, increase the overall precision of the method proposed by Mitra et al. (2014). As per Mitra et al. (2014), the basic idea is to prepare Distributional Thesaurus networks from a corpus that covers old and new time periods, cluster the network around target word to obtain the sense clusters in both the time periods and then do a cluster comparison to find out the sense cluster from the new time period to have a 'birth' sense for the target word which is not present in the old time period. Now, if a target word qualifies as having a new sense ('birth') as per their method, we construct two induced subgraphs of those words that form the cluster corresponding to this 'birth' sense, from the corresponding distributional thesauri (DT) networks of the two time periods. Next, we compare the following three network properties: (i) the edge density (ED), (ii) the structural similarity (SS) and (iii) the average path length (APL) (Turnu, Marchesi, & Tonelli, 2012; Wasserman & Faust, 1994) of the two induced subgraphs from the two time periods. A remarkable observation is that although this is a small set of only three features, for the actual 'birth' cases, each of them has a significantly different value for the later time point and are therefore very discriminative indicators. In fact, the features are so powerful that even a small set of training instances is sufficient for making highly accurate predictions. We pose this problem as a binary classification task where we get the best results for Support Vector Machine (SVM) classifier fed with the fractional change of ED, SS, and APL over time as features.

Preparation of gold standard dataset: In order to evaluate our model, we prepare a gold standard dataset through human annotations. As far as we are aware of the literature, there are no such gold standard datasets (even âæsilver annotationsâg) available, which in turn makes the evaluation task difficult for this particular problem of novel sense detection. This concern is also discussed in detail in the recent surveys (Kutuzov, Øvrelid, Szymanski, & Velldal, 2018; Tahmasebi, Borin, & Jatowt, 2018). In most of the studies dealing with this task, researchers take some example words known to have emerged with new sense and try to model their characteristics over time. On the other hand, the datasets for evaluating tasks like word sense disambiguation or word sense induction do not contain the time information. Considering all these issues, we introduce a carefully prepared gold standard dataset. Note that this gold standard dataset consists of 365 words (nouns extracted from the torso region of word frequency as per Google books corpus), 184 for 1909–1953 vs 2002–2005 and 181 for 1909–1953 vs 2006–2008. This dataset is one of our contributions as well which will help the community to move further in this otherwise difficult task.

Results: Evaluation using this gold standard dataset shows that this classification achieves an overall precision of 0.86 and 0.74 for the two time point pairs over the same set of samples, in contrast with the precision values of 0.32 and 0.23 by the original method. Note that we would like to stress here that an improvement of **more than double** in the precision of novel sense detection that we achieve has the potential to be the new stepping stone in many NLP and IR applications that are sensitive to novel senses of a word.

1.4. Inspection with network embeddings

In addition to exploring the usefulness of network measures obtained from the DT networks, we also investigate the effect of using network representation learning methods. For that purpose we produce network embeddings to map the DT from the network space to the vector space using Deepwalk (Perozzi, Al-Rfou, & Skiena, 2014) and node2vec (Grover & Leskovec, 2016). We propose two measures – Intra-cluster Average Similarity (IAS) and Average Similarity with Target word (AST) computed from the network embeddings, and apply fractional change of these two measures as features in the classifier. We find that using these two features or a combination of these features along with the network features (ED, SS, APL) in the classifier helps to improve precision in some cases but is not able to beat the overall F-measure values of the classifier fed with only network features (ED, SS, APL).

1.5. Inspection with FastText embeddings

We further investigate the usefulness of word embeddings obtained using FastText (Bojanowski, Grave, Joulin, & Mikolov, 2017) trained on the corpus from two different time periods. We use the same two measures as discussed before – Intra-cluster Average Similarity (IAS) and Average Similarity with Target word (AST) but computed using FastText embeddings, and apply fractional change of these two measures as features in the classifier. In this scenario as well, we observe that using these two features or a combination of these features along with the network features (ED, SS, APL) in the classifier helps to improve precision in some cases but is not able to beat the overall F-measure values of the classifier fed with only network features (ED, SS, APL).

1.6. Detecting known shifts

Further, we also investigate the robustness of our approach by analyzing the ability to capture known historical shifts in meaning. Preparing a list of words that have been suggested by different prior works as having undergone sense change, we see that 80% of those words get detected by our approach. We believe that the ability to detect such diachronic shifts in data can significantly enhance various standard studies in natural language evolution and change.

The work presented here is an extension of Jana, Mukherjee, and Goyal (2019). The novel aspects and contributions of this paper with respect to the conference version are (a) it is an extended version of the conference paper with a detailed explanation of the baselines, dataset description, proposed methodology, extensive feature analysis along with illustrative examples and (b) in addition to applying only complex network measures, we also investigate the applicability of network representation learning methods and FastText embedding methods for the task of precise novel sense detection.

2. Related work

Word sense induction and word sense disambiguation are the broad problems that deal with the detection of word senses. Some of the first attempts were made by Mihalcea and Moldovan (1999) where they proposed an approach to automatically generate an arbitrarily large corpus for word senses using the information provided in WordNet and the information gathered from Web using existing search engines. Sahlgren (2002) tried to build a model of semantic knowledge using random indexing, which is able to acquire ambiguous semantic information in an unsupervised fashion from unstructured text data. Recently, Wu and Giles (2015) proposed a multi-prototype model for word representation using Wikipedia, namely SaSA, that could give more accurate sense-specific representation to words with multiple senses. Another stream of literature focuses on inducing senses of words by using clustering approaches with different context representations. Pantel and Lin (2002) presented a clustering algorithm, CBC (Clustering By Committee) to automatically discover senses from the text. In a similar line, Dorow and Widdows (2003) came up with an iterative clustering-based method on the word-relation graph. Bordag (2006) proposed a triplet-based clustering algorithm that instantiated the 'one sense per collocation' observation. Pedersen and Bruce (1998) presented a corpus-based approach to word-sense disambiguation that only requires information that can be automatically extracted from the untagged text. Some of the recent attempts to word sense disambiguation have been made by Raviv, Markovitch, and Maneas (2012). They introduced Concept-Based Disambiguation (CBD), a novel framework that utilizes recent semantic analysis techniques to represent both the context of the word and its senses in a high-dimensional space of natural concepts. Baskaya and Jurgens (2016) presented a new approach to build a semi-supervised WSD system that combines a small amount of sense-annotated data with information from WSI. Some researchers (Erk & Pado, 2007) even observed that the senses of a word are not completely disjoint and attempted to propose a graded representation of word sense.

On the other hand, researchers have also tried to develop data-driven models of language dynamics. One of the first attempts was made by Erk (2006), where the author tried to model this problem as an instance of outlier detection, using a simple nearest neighbor-based approach. Gulordava and Baroni (2011) study the change in the semantic orientation of words using Google book n-grams corpus from different time periods. In another work, Mihalcea and Nastase (2012) attempted to quantify the changes in word usage over time and came up with the intuition that changes in usage frequency and word senses contribute to these differences in usages. In similar lines, Jatowt and Duh (2014) used the Google n-grams corpus from two different time periods and proposed a method to identify semantic change based on the distributional similarity between the word vectors. Tahmasebi et al. (2011) attempted to track sense changes from a newspaper corpus containing articles between 1785 and 1985. Even though the interest for automatic identification of new word senses has grown, the research has been limited by the availability of appropriate evaluation resources. Efforts have been made by Cook, Lau, McCarthy, and Baldwin (2014) to prepare the largest corpus-based dataset of diachronic sense differences. Attempts have been made by Lau, Cook, McCarthy, Newman, and Baldwin (2012) where they first introduced topic modeling based word sense induction method to automatically detect words with emergent novel senses. In subsequent work, Lau et al. (2014) extended this task by leveraging the concept of predominant sense. The first computational approach to track and detect statistically significant linguistic shifts of words has been proposed by Kulkarni et al. (2015). Researchers (Kenter, Wevers, Huijnen, & De Rijke, 2015) also attempted to solve a variant of this problem which deals with monitoring shifts in vocabulary over time. Recently, Hamilton, Leskovec, and Jurafsky (2016b) proposed a method to quantify semantic change by evaluating word embeddings against known historical changes. In another work, Hamilton et al. (2016a) categorized the semantic change into two types and proposed different distributional measures to detect those types. An attempt has also been made to analyze the time-series model of embedding vectors as well as time-indexed self-similarity graphs in order to hypothesize the linearity of semantic change by Eger and Mehler (2016). Dynamic Bayesian model of diachronic meaning change has been proposed by Frermann and Lapata (2016) where they have shown novel sense detection task as one of their applications. The probabilistic language model for time-stamped text data which tracks the semantic evolution of individual words over time has also been tried out (Bamler & Mandt, 2017). Recently, researchers have also tried to investigate the reasons behind word sense evolution and have come up with computational models based on chaining (Ramiro, Srinivasan, Malt, & Xu, 2018). Researchers also attempt to apply dynamic word embeddings as well to detect language evolution. Rudolph and Blei (2018) develop dynamic embeddings, building on exponential family embeddings to capture how the meanings of words change over time. Yao, Sun, Ding, Rao, and Xiong (2018) develop a dynamic statistical model to learn time-aware word vector representation. Researchers also attempt to analyze temporal word analogy by effectively modeling with diachronic word embeddings, provided that the independent embedding spaces from each time period are appropriately transformed into a common vector space (Szymanski, 2017). In a recent study, Di Carlo, Bianchi, and Palmonari (2019) proposed a new heuristic to train temporal word embeddings based on the word2vec model which talks about using atemporal vectors as a reference, i.e., as a compass, when training the representations specific to a given time interval and they evaluated this heuristic for temporal word analogy task.

As per the surveys made in this stream of literature (Kutuzov et al., 2018; Tahmasebi et al., 2018; Tang, 2018), researchers have tried to formulate the task of tracking semantic shifts differently. Given a corpora containing texts from different time periods, some researchers have tried to locate words with different meaning in different time periods (Cook et al., 2014; Lau et al., 2012; Mitra

et al., 2014), some attempted to trace the dynamics of the relationship between the words (Erk & Pado, 2007; Gulordava & Baroni, 2011; Jatowt & Duh, 2014; Mihalcea & Nastase, 2012) whereas some tried to discover the general trends in semantic shifts depicting probable linguistic reasons (Eger & Mehler, 2016; Hamilton et al., 2016a; 2016b; Kulkarni et al., 2015). Recently, a new line of research problem which deals with detecting temporal word analogy has also been attempted (Di Carlo et al., 2019; Szymanski, 2017). Researchers have not only tried to attempt this problem of semantic shift of words from different perspectives with different goals, but they have also tried to formulate solutions to such problems using different methodologies. Starting from simple corpus statistics based approaches (Gulordava & Baroni, 2011; Tahmasebi et al., 2011), researchers have gradually moved toward probabilistic approaches (Bamler & Mandt, 2017; Erk, 2006; Frermann & Lapata, 2016; Mihalcea & Nastase, 2012), topic modelling based approaches (Cook et al., 2014; Lau et al., 2012), and network based approaches (Mitra et al., 2014). Recently, a trend of using dynamic neural embeddings has been observed among the researchers (Di Carlo et al., 2019; Eger & Mehler, 2016; Jatowt & Duh, 2014; Kulkarni et al., 2015; Szymanski, 2017; Yao et al., 2018).

Nevertheless, in majority of these previous works, researchers tried to model the way a word's sense changes over time and validated their models using words known to have undergone sense change. In some attempts, authors tried to find out how or why the semantic shifts happen as well. In contrast to most of these attempts, we pose the problem as detecting the set of words which have come up with a novel sense between old time point (t_{old}) and new time point (t_{new}) with high precision, provided we have large text corpus from both t_{old} and t_{new}. Therefore we point out two such baselines (Lau et al., 2014; Mitra et al., 2014) with a similar objective. We describe these baselines in Section 3.

3. Baselines

In this section, we describe the two baselines that are relevant to our work.

3.1. Baseline 1: (Mitra et al., 2014)

In Mitra et al. (2014), the authors proposed an unsupervised and automated method to identify word sense changes. Their analysis is only focused on noun words. A brief summary of their work is described below for the ease of the readability of this paper.

3.1.1. Datasets and graph construction

The authors used the Google books corpus, consisting of texts from over 3.4 million digitized English books. These books were published between 1520 and 2008, mostly after 1800. The authors constructed distributional thesauri (DT) networks from the Google books syntactic n-grams data (Goldberg & Orwant, 2013). The DT network contains, for each word, a list of words that are similar with respect to their bigram distribution (Riedl & Biemann, 2013). In particular, they first extracted each word and a set of its context features like part-of-speech tag, neighboring set of words, frequency, etc. Next, they calculated the lexicographer's mutual information (LMI) (Kilgarriff, Rychly, Smrz, & Tugwell, 2004)[2] between a word and its features and took the top 1000 ranked features for each word. In the network, each word is a node and there is a weighted edge between a pair of words where the weight of the edge is defined as the number of features that these two words share in common. A snapshot of the DT is shown in Fig. 1. To study word sense changes over time, they divided the dataset across eight time periods; accordingly DT networks for each of these time periods were constructed separately. The basic idea is that if a word undergoes sense change, it can be detected by comparing its senses from two different time periods. The unsupervised method for inducing word senses in each time period is described below.

3.1.2. Unsupervised sense induction

In order to get the induced sense clusters in an unsupervised way, Chinese Whispers algorithm (Biemann, 2006) has been used. The algorithm produces a set of clusters for each target word by decomposing its neighborhood in the DT network. The hypothesis is that different clusters signify different senses of a target word. The clusters for a target word 'float' is shown in Fig. 2. The authors then compare the sense clusters extracted across two different time periods to obtain suitable signals of sense change.

3.1.3. Sense change detection

Let us assume that the Chinese Whispers algorithm is run over DTs corresponding to two different time periods, t_i and t_j. Now, assume that for a given word w, the algorithm gives two different sets of clusters, C_i and C_j, such that m sense clusters are obtained in t_i and n sense clusters are obtained in t_j. Accordingly, let $C_i = s_{i_1}, s_{i_2},..., s_{i_m}$ and $C_j = s_{j_1}, s_{j_2},..., s_{j_n}$, where s_{k_z} denotes z^{th} sense cluster for word w during time interval t_k. There are four types of sense changes that can happen for a target word – *split, join, birth* and *death*. These sense changes are defined below.

- **split**: A sense cluster s_{i_z} in t_i splits into two (or more) sense clusters, $s_{j_{p_1}}$ and $s_{j_{p_2}}$ in t_j.
- **join**: Two sense clusters $s_{i_{z_1}}$ and $s_{i_{z_2}}$ in t_i join to make a single cluster s_{j_p} in t_j.

[2] $$LMI(word, feature) = f(word, feature)*\log_2(f(word, feature)/f(word)*f(feature))$$

, where $f()$ measures the frequency.

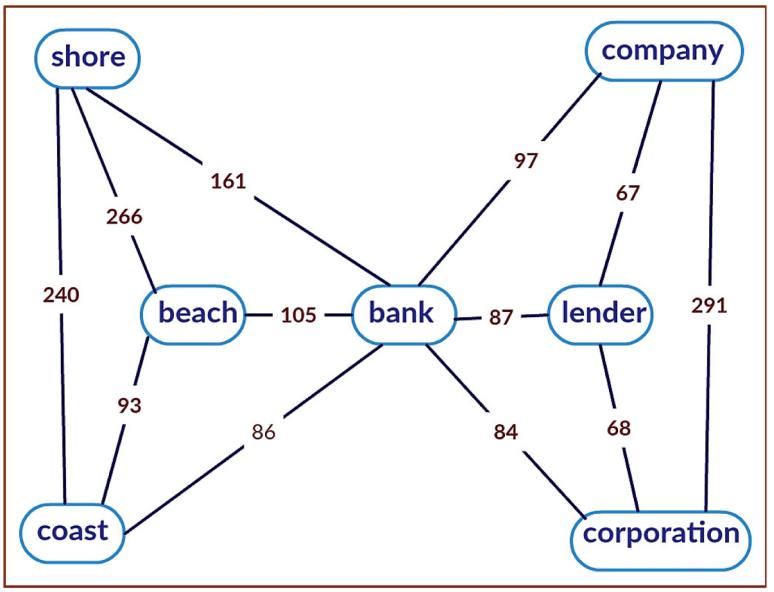


Fig. 1. A sample snapshot of Distributional Thesaurus Network from the time period 2002–2005 where each node represents a word and the weight of the edge is defined as the number of context features that these two words share in common. Here the word 'bank' with some top distributionally similar words and the connections among them are shown.

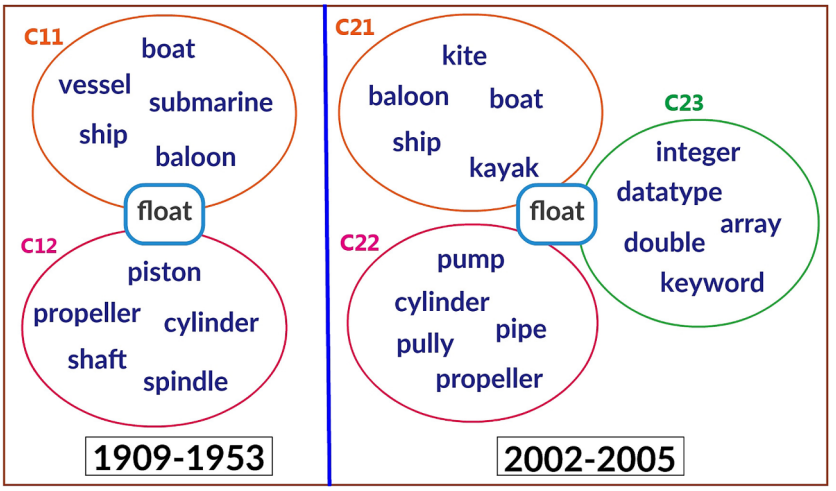


Fig. 2. Chinese Whisper clusters for the target word 'float' extracted from Google books syntactic n-gram corpus of both the time periods (1909–1953 and 2002–2005). A new sense of the word 'float' has emerged with the 'programming' related new cluster (C23) in 2002–2005.

- **birth**: A new sense cluster s_{j_p} appears in t_j, which was absent in t_i.
- **death**: A sense cluster s_{i_z} in t_i dies out and does not appear in t_j.

A sense cluster is considered as 'birth' if at least 80% words of that cluster are novel, i.e., they do not appear in any of the clusters of the old time periods. For example, in Fig. 2, the programming related cluster of the target word 'float' represents a 'birth' sense. For split, each split cluster should have at least 30% words of the source cluster and the total intersection of all the split clusters should be $>$ 80%. For join and death, the same parameters are used with the interchange of the source and the target clusters.

Note that, as our main focus is to detect novel sense of a word, we are concerned with only 'birth' cases for our study.

3.1.4. Multi-stage filtering

The authors then apply multi-stage filtering in order to obtain meaningful candidate words.

Stage 1: They apply Chinese Whisper three times over the two different time periods. They obtain the candidate word lists using their algorithm for the three runs, then take the intersection to output those words and clusters, which came up in all the three runs.

Being non-deterministic in nature, the Chinese Whisper algorithm might produce different clustering in different runs. Therefore running it multiple times and taking an intersection is helpful to reduce the effect of non-determinism.

Stage 2: As they focus only on nouns, they keep the candidate words tagged with 'NN' or 'NNS'.

Stage 3: They sort the target words based on their frequency counts and consider only the middle 60% of the list which is the most informative part for this type of analysis. Note that, the authors remove the words in the low-frequency range as there may not be sufficient evidence in the dataset to detect a sense change and rare words usually only have a single sense. On the other hand, words in the high-frequency range tend to be less topic-oriented and thus, appear in very different contexts even when conveying the same (mostly abstract) sense (Kwong, 1998; Luhn, 1958). For evaluation, the authors selected 49 candidate 'birth' words from a total of 2789 candidate 'birth' words while comparing 1909–1953 DT with the 2002–2005 DT. Using manual evaluation, 31 words were found to be true positives and 18 words were false positives. In our work, we take these 49 candidate words and show that network features can be useful to discriminate the true positives from the false positives.

3.2. Baseline 2: Lau et al. (2014)

The authors proposed an unsupervised approach based on topic modeling for sense induction and showed novel sense identification as one of its applications. For a candidate word, Hierarchical Dirichlet Process (Teh, Jordan, Beal, & Blei, 2006) is run over a corpus containing occurrences of that word to induce topics. The induced topics are represented as word multinomials, and are expressed by the top-N words in descending order of conditional probability. Each topic is represented as a sense of the target word. The words having the highest probability in each topic represent the sense clusters. The authors treated the novel sense detection task as identifying those sense clusters, which did not align with any of the recorded senses in a sense repository. They used Jensen-Shannon (JS) divergence measure to compute the alignment between a sense cluster and a synset. First, they computed JS divergence between the multinomial distribution (over words) of the topic cluster and that of the synset, and converted the divergence value into a similarity score. Similarity between topic cluster t_j and synset s_i is defined as

$$sim(t_j, s_i) = 1 - JS(T||S) \tag{1}$$

where T and S are the multinomial distributions over words for topic t_j and synset s_i, respectively, and $JS(X\|Y)$ is the Jensen-Shannon divergence for the distribution X and Y. Since we define novel senses while comparing sense clusters across two time periods, we use the same JS measure to detect novel sense of a target word. A sense cluster in the newer time period denotes a new sense ('birth') only if its maximum similarity with any of the clusters in older time period is below a threshold, which we have set to 0.35 based on empirical observation.

In Mitra et al. (2014), the authors reported a total of 2789 candidate 'birth' words while comparing 1909–1953 DT with 2002–2005 DT and 2468 candidate 'birth' words while comparing 1909–1953 DT with 2006–2008 DT. We take all these reported 'birth' cases as target words and apply our approach to improve the quality of the detected 'birth' senses. We have also followed the procedure described above for Lau et al. (2014) over the same set of candidate words. Then we take 100 random samples from each of these two separate set-ups(1909–1953 vs 2002–2005 and 1909–1953 vs 2006–2008) and compare all three approaches in terms of precision-recall measure over this set using manual evaluation of the reported results.

4. Proposed network-centric approach

Mitra et al. (2014) manually evaluated 49 candidate 'birth' words from a total of 2789 candidate 'birth' words while comparing 1909–1953 DT with the 2002–2005 DT among which 31 words were found to be having come up with novel sense and 18 words are not having novel sense. We first study these 49 candidate 'birth' words and show that network features can be useful to discriminate the true positives from the false positives. For each of these candidate words w, we take the 'birth' cluster from 2002 to 2005, which is represented by a set of words S. According to our hypothesis, if the words in set S together represent a new sense for w in 2002–2005 which is not present in 1909–1953, the network connection among these words (including w) would be much stronger in the 2002–2005 DT than the 1909–1953 DT. The strength of this connection can be measured if we construct induced subgraphs of S from the two DTs and measure the network properties of these subgraphs; the difference would be more prominent for the actual 'birth' cases (true positives) than for the false 'birth' signals (false positives). Note that by definition, the nodes in an induced subgraph from a DT will be the words in S and there will be an edge between two words if and only if the edge exists in the original DT; we ignore the weight of the edge henceforth. Thus, the difference between the two subgraphs (one each from the older and newer DTs) will only be in the edge connections. Fig. 3 shows one true positive ('register') and one false positive ('quotes') word from the set of 49 words and shows the induced subgraphs obtained by a subset of their 'birth' clusters across the two time periods. Note that, the target words are not present in the figure. This figure basically depicts, how the network connections among the words in the birth cluster (signifying the sense of target word) change over time. We can clearly see that connections among the words in S are much stronger in the newer DT than in the older ones in the case of 'registers', indicating the emergence of a new sense. In the case of 'quotes', however, the connections are not very different across the two time periods. We choose three *cohesion indicating* network properties, (i) the edge density, (ii) the structural similarity and (iii) the average path length, to capture this change.

Let $S = \{w_1, w_2, ..., w_n\}$ be the 'birth' cluster for w. Once we construct a graph induced by S from the DT, these network properties are measured as follows:

Edge Density (ED): ED is given by

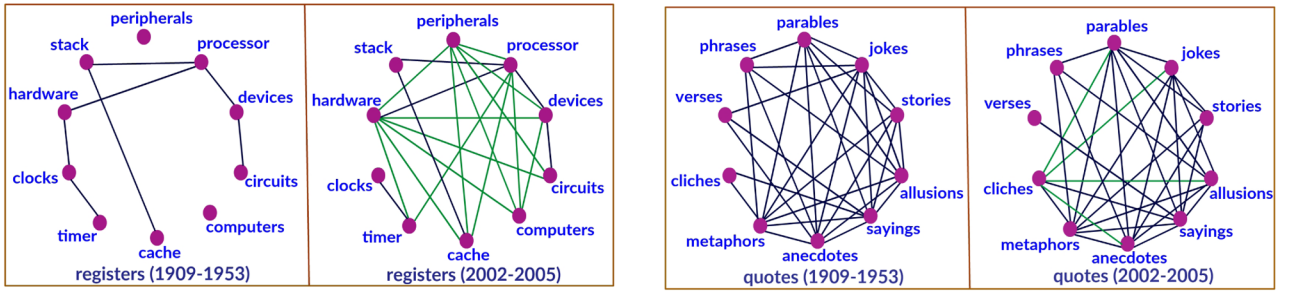


Fig. 3. Induced subgraphs of the 'birth' clusters of 'registers' and 'quotes' for the two time periods (1909–1953 and 2002–2005). It shows that edge connections among the neighbors of 'registers' have increased significantly over time which leads to the emergence of 'technology' related sense of 'registers' whereas the connections among the neighbors of 'quotes' are almost same over time, indicating non-emergence of any novel sense.

Table 1
The network properties of the induced subgraphs of a true positive ('registers') and a false positive ('quotes') for the time periods 1909–1953 (t_1) and 2002–2005 (t_2). The fractional changes (Δ) in network properties are significantly higher for 'registers' compared to 'quotes'.

Word	ED (t_1)	ED (t_2)	SS (t_1)	SS (t_2)	APL (t_1)	APL (t_2)	Δ (ED, SS, APL)
registers	0.108	0.546	0.076	0.516	1.9	1	4.045, 5.771, -0.9
quotes	0.858	0.833	0.835	0.622	1.72	1	-0.029, -0.255, -0.72

$$ED = N_a/N_p \tag{2}$$

where N_a denotes the number of actual edges between $w_1, w_2, \ldots, w_n$ and N_p denotes the maximum possible edges between these, i.e., $\frac{n(n-1)}{2}$. This measure indicates how densely the neighbours of the target word are connected among themselves.

Structural Similarity (SS): For each pair of words (w_i, w_j) in the cluster S, the structural similarity $SS(w_i, w_j)$ is computed as:

$$SS(w_i, w_j) = \frac{N_c}{\sqrt{deg(w_i)*deg(w_j)}} \tag{3}$$

where N_c denotes the number of common neighbors of w_i and w_j in the induced graph and $deg(w_k)$ denotes the degree of w_k in the induced graph, for $k = i, j$. The average structural similarity for the cluster S is computed by averaging the structural similarity of all the word pairs. This measure indicates the cohesiveness among the neighborhood of the target word.

Average Path Length (APL): To compute average path length of S, we first find the shortest path length between w and the words w_i, in the induced graph of S. Let spl_i denote the shortest path distance from w to w_i. The average path length is defined as:

$$APL = \sum_i spl_i/n \tag{4}$$

where n is the number of words in the cluster S. This measure captures how the distance between the target word and the words in the birth cluster changes over time in the network. If the words in the birth cluster come close over time, it signifies that the target word is emerging with a sense captured by the birth cluster.

Table 1 notes the values obtained for these network properties for the induced subgraphs of the reported 'birth' clusters for 'registers' and 'quotes' across the two time periods. The fractional changes observed for the three network properties show a clear demarcation between the two cases. Fractional change (Δ) of any network measure P is defined as,

$$\Delta(P) = (P(t_2) - P(t_1))/P(t_1) \tag{5}$$

where t_1 and t_2 are old and new time periods respectively. The change observed for the 'birth' cluster of 'registers' is significantly higher than that in 'quotes'[3].

We now compute these parameter values for all the 49 candidate cases. The mean values obtained for the true positives (TP) and false positives (FP) are shown in Table 2. The findings are consistent with those obtained for 'registers' and 'quotes'.

We, therefore, use the fractional changes in the three network properties over time as three features to classify the remaining candidate 'birth' words into true positives (actual 'birth') or false positives (false 'birth'). We use different classifiers like Naive Bayes, Support Vector Machine (SVM), Random Forest, etc. for this purpose and report the result of SVM, which was the best performing classifier.

4.1. Investigation using network representation learning approaches

In addition to the proposed three metrics inspired by network science, we also explore the applicability of network representation learning methods in this task. Recall that, we have the DT networks both from the old (DT_{old}) and the new (DT_{new}) time periods. First, we apply network representation learning methods to obtain continuous feature representations for nodes in networks for both the time periods. As discussed in Section 4, for each candidate word w, we have the 'birth' cluster from 2002 to 2005, which is represented by a set of words S. Intuitively, if the words in set S together represent a new sense for w in 2002–2005 which is not present in 1909–1953, the relative similarity in the vector space between the words in S among themselves as well as with the target words would be much higher in the 2002–2005 DT in comparison to the 1909–1953 DT. In order to measure these similarities, we propose the following two metrics.

Intra-cluster Average Similarity (IAS): To compute this metric for S, we first find the pairwise cosine similarity ($CSim$) between the words in S. Then we consider the average of these cosine similarities. IAS is defined as:

$$IAS = \frac{2}{|S|*|S-1|}* \sum_{w_i, wj \in S, i<j} CSim(w_i, w_j) \tag{6}$$

Average Similarity with Target word (AST): To compute this metric for S, we first find the cosine similarity ($CSim$) between the

[3] As we have taken the 'birth' clusters from new time period (t_2), the words in the clusters are the direct neighbors of the target word always resulting in average path length of 1 in t_2

Table 2
Mean values of the network properties of the induced subgraphs of 31 true positives and 18 false positives for the time periods 1909–1953 (t_1) and 2002–2005 (t_2). The mean fractional changes (Δ) in network properties are significantly higher for the true positives (TP) as compared to the false positives (FP) which indicate that the words emerged with new senses have undergone a drastic change in network connections in their neighborhood over time.

Word	ED (t_1)	ED (t_2)	SS (t_1)	SS (t_2)	APL (t_1)	APL (t_2)	Δ (ED, SS, APL)
TP	0.34	0.772	0.311	0.647	1.941	1	2.388, 4.654, -0.941
FP	0.576	0.828	0.574	0.681	1.828	1	0.747, 0.507, -0.828

target word w and the words in S. Then, we consider the average of these cosine similarities. AST is defined as:

$$AST = \frac{1}{|S|} * \sum_{w_i \in S} CSim(w, w_i) \tag{7}$$

We use the fractional change (Δ) (as described in Eq. (5)), of these two measures as features to classify the candidate 'birth' words into true positives (actual 'birth') or false positives (false 'birth').

Now, from the DT network of each time period, in order to prepare the vector representations for each node, we explore two state-of-the-art network representation learning models as discussed below.

Deepwalk: Deepwalk (Perozzi et al., 2014) learns social representations of a graph's vertices by modeling a stream of short random walks. Social representations signify latent features of the vertices that capture neighborhood similarity and community membership. Using local information from the truncated random walks as input, this method learns a representation that encodes structural regularities.

node2vec: node2vec (Grover & Leskovec, 2016) is an algorithmic framework for scalable feature learning in networks, that maximizes the likelihood of preserving network neighborhoods of nodes in a d-dimensional feature space. This algorithm can learn representations that organize nodes based on their network roles and/or communities they belong to by developing a family of biased random walks, which efficiently explore diverse neighborhoods of a given node.

4.2. Investigation using fastText embeddings

We further investigate the usefulness of embeddings obtained directly from corpus compared to the embeddings obtained using network embeddings from network representation of corpus. We obtain fastText embeddings (Bojanowski et al., 2017) from both the Google book corpus of old and new time periods. We use the same metrics (IAS, AST) as proposed in Section 4.1 as features to be plugged into a classifier, but in order to compute those two metrics, we use the fastText embeddings of words in place of network embeddings (Deepwalk, node2vec). We use the vector dimension of 128 to be consistent with the analysis using network embeddings.

5. Experimental results

For experimental evaluation, we start with the 'birth' cases reported by Mitra et al. (2014) – 2740 cases (after removing the 49 cases used in training) for 1909–1953 - 2002–2005 (T_1)and 2468 cases for 1909–1953 - 2006–2008 (T_2). We first discuss how the gold standard dataset is prepared in Section 5.1. In Section 5.2, we run Lau et al. (2014) over these birth cases to detect 'novel' sense as per Lau et al.'s algorithm. Separately, we also apply the proposed SVM classification model as a filtering step to obtain 'filtered birth' cases. This helps in designing a *comparative evaluation* of these algorithms as follows. From both the time period pairs (T_1 and T_2), we take 100 random samples from the birth cases reported by Mitra et al. (2014) and get them evaluated against gold standard dataset prepared by human annotators. For the same 100 random samples, we now use the outputs of Lau et al. (2014) and the proposed approach and estimate the precision as well as recall of these. Note that, for our proposed SVM based model, even though we use these random 100 samples from both T_1 and T_2 for testing, the training set is fixed to the 49 birth cases taken from T_1 provided by Mitra et al. (2014). This comparative evaluation shows, how much we achieve by applying our proposed model on top of the approach of Mitra et al. (2014). We also investigate how good our model performs if we apply it independent of Mitra et al. (2014). Next, we do an extensive feature analysis to explore the importance of each of the network measures. In Section 5.3 we analyze the usefulness of features obtained using Deepwalk and node2vec independently as well as in combination with the proposed network features. Next, in Section 5.4 we do the same analysis using word embeddings obtained using fastText. As a final step, we perform an error analysis of our proposed approach in Section 5.5.

5.1. Gold standard preparation

As far as we are aware of the literature, there is a scarcity of gold standard datasets, which in turn makes the evaluation task difficult for this particular problem of novel sense detection. This concern is also discussed in detail in the recent survey papers (Kutuzov et al., 2018; Tahmasebi et al., 2018). Therefore, we prepare a gold standard dataset using human annotations and perform all the evaluations against this gold standard dataset. Each of the candidate words is judged by three evaluators. These evaluators are graduate/post-graduate students, having a good background in Natural Language Processing and well versed with the

English language. They are unaware of each other, making the annotation process completely blind and independent. Evaluators are shown the detected 'birth' cluster from the newer time period and all the clusters from the older time period. They are asked to make a binary judgment as to whether the 'birth' cluster indicates a new sense of the candidate word, which is not present in any of the sense clusters of the previous time point.[4] Majority decision is taken in the disagreement. In total, we prepared annotations for a set of as large as **365** words (only nouns taken from Distributional Thesaurus)[5] which we believe is significant given the tedious manual judgment involved. In this process of manual annotation, we obtain an inter-annotator agreement (Fleiss' κ (Fleiss, 1971)) of **0. 745**, which is *substantial* (Viera, Garrett et al., 2005). Table 3 shows three example words from T_1, their 'birth' clusters as reported in Mitra et al. (2014) and the manual evaluation result. The first three cases belong to computer or technology related sense of 'sender', 'directories' and 'float', which were absent from time point 1909–1953. On the other hand, the 'birth' clusters of 'celebrity' and 'quiz' represent an old sense which was also present in 1909–1953. Similarly, Table 4 shows manual evaluations results for three example cases, along with their novel sense as captured by Lau et al. (2014). This gold standard dataset is also one of our significant contributions and we make it publicly available to facilitate further research.

5.2. Comparative evaluation

Only 32 and 23 words out of the 100 random samples from two time point pairs are evaluated to be actual 'birth's, respectively, thus giving precision scores of 0.32 and 0.23 for Mitra et al. (2014). Evaluation results for the same set of random samples after applying the approach outlined in Lau et al. (2014) are presented in Table 5. Since the reported novel sense cluster can in principle be different from the 'birth' sense reported by the method of Mitra et al. (2014) for the same word, we get the novel sense cases manually evaluated by 3 annotators (42 and 28 cases for the two time periods, respectively). Note that for these 100 random samples (that are all marked 'true' by Mitra et al. (2014)), it is possible to find an upper bound on the recall of Lau et al. (2014)'s approach automatically. While the low recall might be justified because this is a different approach, even the precision is found to be in the same range as that of Mitra et al. (2014).

Table 6 presents the evaluation results for the same set of 100 random samples after using the proposed SVM filtering. We see that the filtering using SVM classification improves the precision for both the time point pairs (T_1 and T_2) significantly, boosting it from the range of 0.23-0.32 to 0.74-0.86. Note that, as per our calculations, indeed the recall of Mitra et al. (2014) would be 100% (as we are taking random samples for annotation from the set of reported 'birth' cases by Mitra et al. (2014) only). Even then Mitra et al. (2014)'s F-measure ranges from 0.37-0.48 while ours is 0.67-0.68. Table 7 represents some of the examples which were declared as 'birth' by Mitra et al. (2014) but SVM filtering correctly flagged them as 'false birth'. The feature values in the third column clearly show that the network around the words in the detected 'birth' clusters did not change much and therefore, the SVM approach could correctly flag these. Considering the small training set (49 reported 'birth cases' by Mitra et al. (2014)), the results are highly encouraging. We also obtain decent recall values for the two time point pairs, giving an overall F-measure of 0.67-0.68.

5.2.1. Feature analysis

We, therefore, move onto further feature analysis of the proposed approach. To validate the usefulness of all three proposed features, we first check the Pearson's correlation among the three features and then perform feature leave-one-out experiments. Table 8 represents the feature correlation matrix which shows that the three features are not significantly correlated. Next, we observe the results of feature leave-one-out experiment for T_1 and T_2 in Table 13 and Table 14 respectively. We find that the F1-score drops as we leave out one of the features. While {*ED, SS*} turns out to be the best for precision, {*SS, APL*} gives the best recall. Table 11 provides three examples to illustrate the importance of using all three features. For the word 'newsweek', using {*ED, APL*} and for the word 'caring', using {*ED, SS*} could not detect those as 'birth'. Only when all the three features are used, these cases are correctly detected as 'birth'. Edge density, on the other hand, is very crucial for improving precision. For instance, when only {*SS, APL*} are used, words like 'moderators' are wrongly flagged as 'true'. Such cases are filtered out when all the three features are used.

5.2.2. Extensive analysis of our proposed approach

We first take 60 random samples each from the filtered 'birth' cases reported by the SVM filtering for the two time period pairs, T_1 (from 318 cases) and T_2 (from 329 cases). The precision values of this evaluation are found to be **0.87** (52/60) and **0.75** (45/60) respectively, quite consistent with those reported in Table 6. We do another experiment in order to estimate the performance of our model for detecting novel sense, independent of the method of Mitra et al. (2014). We take 100 random words from the two time point pairs (T_1 and T_2), along with all the induced clusters from the newer time period and run the proposed SVM filtering approach to flag the novel 'birth' senses. According to our model, for T_1 and T_2 respectively, 13 out of predicted 24 words and 13 out of 21 predicted words are flagged to be having novel sense achieving precision values of **0.54** and **0.62** on manual evaluation, which itself is quite decent. Note that, for some cases, multiple clusters of a single word have been flagged as novel senses and we observe that these clusters hold a similar sense. For both of these experiments we use the same set of 49 reported 'birth cases' by Mitra et al. (2014) to train the SVM classifier (Tables 9 and 10).

[4] An anonymized sample evaluation page can be seen here: https://kwiksurveys.com/s/7TfSoYF2

[5] 100 + 60 + 24 for T_1 and 100 + 60 + 21 for T_2

Table 3
Example 'birth' clusters reported in Mitra et al. (2014) and manual evaluation. For each word, 'birth' cluster represents the words in the Chinese Whisper cluster which is flagged as a new sense by the model.

Word	'birth' cluster	Manual Evaluation
sender	server, handler, proxy, host, ···	**Yes**, Network transfer related sense
directories	file, cache, folder, repository, ···	**Yes**, Digital storage related sense
float	integers, type, array, double, ···	**Yes**, Data type related sense
celebrity	actor, musician, hero, politician, athlete, ···	**No**
quiz	contest, prize, contests, games, ···	**No**

Table 4
Example novel senses as per Lau et al. (2014) and manual evaluation. 'Novel sense' represents the list of words which indicates a new sense as per Lau et al. (2014).

Word	**Novel sense**	**Manual Evaluation**
adapter	connectivity, tools, system, address, ···	**Yes**, Technology related sense
clicks	link, page, user, visitor, ···	**Yes**, Internet related sense
pampering	advice, expert, condition, delicate, ···	**No**

Table 5
Evaluation of the approach presented in Lau et al. (2014) with accuracy for 100 random samples for T_1 (1909–1953 vs 2002–2005) and T_2 (1909–1953 vs 2006–2008).

Time-	Lau et al. (2014)			
point	# Novel senses	Precision	Recall	F-measure
T_1	1189	0.21	0.28	0.24
T_2	787	0.28	0.35	0.31

Table 6
Evaluation of the SVM-based filtering with accuracy reported for 100 random samples for T_1 (1909–1953 vs 2002–2005) and T_2 (1909–1953 vs 2006–2008).

Time-	SVM filtering			
point	# birth cases	Precision	Recall	F-measure
T_1	318	**0.86**	0.56	0.68
T_2	329	**0.74**	0.61	0.67

Table 7
Example cases, which Mitra et al. (2014) declared as true 'birth' but SVM filtering correctly filtered.

Word	'birth' cluster	Δ(ED, SS, APL)
mantra	dharma, deity, guru, deities, ···	0.2, −0.03, −0.3
slogan	motto, initials, symbol, trademark, ···	−0.03, −0.26, −0.72
teddy	dolls, puppies, pet, mama, ···	−0.016, −0.25, −0.83

Table 8
The Pearson's correlation matrix for the three complex network measures (ED, SS, APL).

	ED	SS	APL
ED	1	0.243	−0.474
SS	0.243	1	−0.026
APL	−0.474	−0.026	1

5.3. Evaluation of features obtained from network embeddings

5.3.1. Deepwalk

We explore the effect of fractional change of IAS and AST over time by plugging them as features into the SVM classifier. We first try to find out the best hyper-parameter settings for the feature combination of IAS and AST by grid search. We try out different

Table 9
Feature leave-one-out results (T_1).

Features used	Precision	Recall	F-measure
Δ(ED, SS)	0.85	0.53	0.65
Δ(ED, APL)	0.84	0.5	0.62
Δ(SS, APL)	0.81	0.56	0.66
Δ(ED, SS, APL)	**0.86**	0.56	**0.68**

Table 10
Feature leave-one-out results (T_2).

Features used	Precision	Recall	F-measure
Δ(ED, SS)	0.72	0.56	0.63
Δ(ED, APL)	0.73	0.6	0.66
Δ(SS, APL)	0.66	0.61	0.63
Δ(ED, SS, APL)	**0.74**	0.61	**0.67**

Table 11
Example cases to show the utility of all the features (T_1). The true positive cases like 'newsweek' and 'caring' get successfully detected whereas 'moderators' gets successfully detected as false positive if all the three features are considered together.

Word	'birth' cluster	Δ(ED, SS, APL)
newsweek	probation, counseling, ···	0.82, 1.58, −1.3
caring	insightful, wise, benevolent, ···	0.2, 0.13, −2.21
moderators	correlate, function, determinant, ···	0.56, 0.44, −1.78

Table 12
Comparison of performance of (Δ(IAS, AST)) with varying hyper-parameters. All the measures are for T_1 and T_2 when we use Deepwalk for producing network embeddings. d - dimension of vectors, nrw - number of random walks to start at each node, lrw - length of random walk starting at each node.

Time-point	d	lrw	nrw	Precision	Recall	F-measure
	300	40	10	0.82	**0.56**	**0.67**
	128	40	10	**0.85**	0.53	0.65
	80	40	10	0.32	1	0.48
	128	80	10	0.77	0.53	0.63
T_1	128	20	10	0.53	0.62	0.57
	128	10	10	0.32	1	0.48
	128	40	20	0.81	0.53	0.64
	300	40	10	0.56	0.56	0.56
	128	40	10	0.61	**0.61**	0.61
	80	40	10	0.23	1	0.37
	128	80	10	**0.64**	**0.61**	**0.62**
T_2	128	20	10	0.73	0.48	0.58
	128	10	10	0.69	0.48	0.56
	128	40	20	0.62	0.56	0.59

values of the primary hyper-parameters (dimension of vectors, number of random walks to start at each node, length of random walk starting at each node) to obtain the best settings. We experiment with dimension of vectors: (d) = {80, 128, 300}; number of random walks to start at each node (nrw) = {10, 20}; length of random walk starting at each node (lrw) = {10, 20, 40, 80}. From the results presented in Table 12, we see for $d = 128$, $nrw = 10$, $lrw = 40$ we get the best consistent performance over T_1 and T_2 and hence we continue with this hyper-parameter setting (**BDHS**) for further analysis.

We observe from the first rows of Tables 13 and 14, that for both the time period pairs (T_1 and T_2, respectively), using only features obtained using Deepwalk (with best possible hyper-parameter setting) - IAS and AST, does not help improving the performance when compared to the settings in which only network measures are used (Table 6). We next try to combine the simple network features (ED, SS and APL) with IAS and AST as well in different combinations and present the results in Tables 13 and 14 for T_1 and T_2, respectively. We observe that even though the combination of ED, SS, APL, AST gives the best overall performance beating the

Table 13
Comparison of performance of simple complex network measures (Δ(ED, SS, APL)) vs features computed using Network Embedding (**Deepwalk**) (Δ(IAS, AST)). It also shows the performance of the combination of two types of features. All the measures are for $\boldsymbol{T_1}$. The hyper-parameter used for Deepwalk training - dimension of vectors (d) = 128; number of random walks to start at each node (nrw) = 10; length of random walk starting at each node (lrw) = 40.

Features used	Precision	Recall	F-measure
Δ(IAS, AST)	0.85	0.53	0.65
Δ(ED, SS, APL, IAS)	0.85	0.53	0.65
Δ(ED, SS, APL, AST)	**0.91**	**0.62**	**0.74**
Δ(ED, SS, APL, IAS, AST)	0.9	0.59	0.72

Table 14
Comparison of performance of simple complex network measures (Δ(ED, SS, APL)) vs Features computed using Network Embedding (**Deepwalk**) (Δ(IAS, AST)). It also shows the performance of combination of two types of features. All the measures are for $\boldsymbol{T_2}$. The hyper-parameter used for Deepwalk training - dimension of vectors (d) = 128; number of random walks to start at each node (nrw) = 10; length of random walk starting at each node (lrw) = 40.

Features used	Precision	Recall	F-measure
Δ(IAS, AST)	0.61	0.61	0.61
Δ(ED, SS, APL, IAS)	**0.74**	**0.61**	**0.67**
Δ(ED, SS, APL, AST)	0.62	0.56	0.59
Δ(ED, SS, APL, IAS, AST)	0.6	0.52	0.56

Table 15
Comparison of performance of simple complex network measures (Δ(ED, SS, APL)) vs combination with features computed using network embedding (**node2vec**) (Δ(IAS, AST)). All the measures are for $\boldsymbol{T_1}$. The default hyper-parameters used for node2vec training - dimension of vectors (d) = 128; number of random walks to start at each node (nrw) = 10; length of random walk starting at each node (lrw) = 40. Only the last row shows the performance of the best combination of complex network features and network embedding (**node2vec**) features (Δ(ED, SS, APL, AST)) with node2vec trained with the best hyper-parameter setting of Deepwalk (BDHS).

Features used	Precision	Recall	F-measure
Δ(ED, SS, APL)	0.86	**0.56**	**0.68**
Δ(ED, SS, APL, IAS)	0.85	0.53	0.65
Δ(ED, SS, APL, AST)	0.89	0.5	0.64
Δ(ED, SS, APL, IAS, AST)	0.89	0.5	0.64
Δ(ED, SS, APL, AST) [BDHS]	**0.9**	0.53	0.67

performance of the model using only network measures (ED, SS, APL) for T_1, it shows inconsistent performance for T_2 producing poorer performance while compared with the model where only network measures (ED, SS, APL) are used.

5.3.2. node2vec

We further investigate the usefulness of IAS and AST while those are computed from the network embeddings obtained from node2vec. The results for both time period pairs (T_1 and T_2) are presented in Table 15 and Table 16, respectively. We observe that even though appending AST to (ED, SS, APL) improves the precision for both the time period pairs (T_1 and T_2), it leads to decrease of F-measure while compared with using only the network measures. We obtain these results for the default hyper-parameter settings of node2vec as follows: number of random walks to start at each node = 10; length of random walk starting at each node = 80; skipgram window size = 10; Inout = 1; Return = 1. We also try with the hyper-parameter settings for which we get the best results for Deepwalk (**BDHS**) by setting skipgram window size = 5 and the length of random walk starting at each node = 40, the results of which are presented in the last rows of Tables 15 and 16 for T_1 and T_2, respectively. We observe that even in this hyper-parameter setup, node2vec produces poor F-measure score compared to the model using only complex network measures.

We see in all the experiments using network embeddings (for both Deepwalk and node2vec) that no combination of features produces the best performance consistently for both T_1 and T_2. As we are mapping network to vector space, there is a chance that we are missing some information which leads to inconsistent feature values for T_1 and T_2, causing inconsistent performances. On the other hand, simple network measures perfectly capture the change in the networks which leads to consistent feature values producing a consistent performance for T_1 and T_2.

Table 16
Comparison of performance of simple complex network measures (Δ(ED, SS, APL)) vs combination with features computed using Network Embedding (**node2vec**) (Δ(IAS, AST)). All the measures are for T_2. The hyper-parameters used for node2vec training - dimension of vectors (d) = 128; number of random walks to start at each node (nrw) = 10; length of random walk starting at each node (lrw) = 40. Only the last row shows the performance of the best combination of complex network features and network embedding (**node2vec**) features (Δ(ED, SS, APL, AST)) with node2vec trained with the best hyper-parameter setting of Deepwalk (BDHS).

Features used	Precision	Recall	F-measure
Δ(ED, SS, APL)	**0.74**	**0.61**	**0.67**
Δ(ED, SS, APL, IAS)	**0.74**	**0.61**	**0.67**
Δ(ED, SS, APL, AST)	0.76	0.56	0.65
Δ(ED, SS, APL, IAS, AST)	0.75	0.52	0.62
Δ(ED, SS, APL, AST) [BDHS]	0.7	**0.61**	0.65

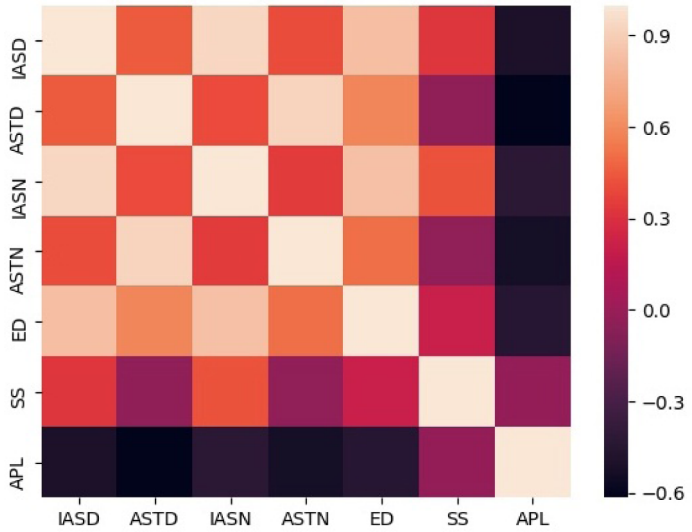


Fig. 4. The heatmap of Pearson's correlation among three network measures (ED, SS, APL), two Deepwalk measures (IASD, ASTD) and two node2vec measures (IASN, ASTN).

Table 17
Comparison of performance of the complex network measures (Δ(ED, SS, APL)) vs combination with features computed using **Deepwalk** (Δ (IASD, ASTD)) and **node2vec** (Δ (IASN, ASTN)). All the measures are for T_1.

Features used	Precision	Recall	F-measure
Δ(ED, SS, APL)	0.86	0.56	0.68
Δ(IASD, ASTD, IASN, ASTN)	0.57	**0.62**	0.6
Δ(ED, SS, APL, IASD, ASTD)	**0.9**	0.59	**0.72**
Δ(ED, SS, APL, IASD, ASTD, IASN)	**0.9**	0.53	0.67
Δ(ED, SS, APL, IASD, ASTD, ASTN)	0.89	0.5	0.64
Δ(ED, SS, APL, IASN, ASTN)	0.89	0.5	0.64
Δ(ED, SS, APL, IASN, ASTN, IASD)	0.89	0.5	0.64
Δ(ED, SS, APL, IASN, ASTN, ASTD)	0.89	0.5	0.64
Δ(ED, SS, APL, IASN, ASTN, IASD, ASTD)	0.88	0.47	0.61

5.3.3. Combination of Deepwalk and node2vec

So far we have discussed the usefulness of features obtained using Deepwalk and node2vec independently. In this section, we try to further combine the two features (IAS, AST) obtained both using Deepwalk and node2vec (naming them IASD, ASTD and IASN, ASTN respectively) along with simple network measures (ED, SS, APL). To validate the usefulness of all seven features, we first check the Pearson's correlation among the seven features, the heatmap of which is presented in Fig. 4. Then we perform feature leave-one-out experiments for network embedding features with the proposed three network measures taken into consideration for all the combinations. From the results presented in Tables 17 and 18, we do not find any single feature combination which beats the

Table 18
Comparison of performance of simple complex network measures (Δ(ED, SS, APL)) vs combination with features computed using **Deepwalk** (Δ(IASD, ASTD)) and **node2vec** (Δ(IASN, ASTN)). All the measures are for T_2.

Features used	Precision	Recall	F-measure
Δ(ED, SS, APL)	0.74	0.61	**0.67**
Δ(IASD, ASTD, IASN, ASTN)	0.51	**0.87**	0.64
Δ(ED, SS, APL, IASD, ASTD)	0.6	0.52	0.56
Δ(ED, SS, APL, IASD, ASTD, IASN)	0.67	0.52	0.58
Δ(ED, SS, APL, IASD, ASTD, ASTN)	0.63	0.52	0.57
Δ(ED, SS, APL, IASN, ASTN)	**0.75**	0.52	0.62
Δ(ED, SS, APL, IASN, ASTN, IASD)	**0.75**	0.52	0.62
Δ(ED, SS, APL, IASN, ASTN, ASTD)	0.63	0.52	0.57
Δ(ED, SS, APL, IASN, ASTN, IASD, ASTD)	0.71	0.52	0.6

Table 19
Comparison of performance of simple complex network measures (Δ(ED, SS, APL)) vs combination with features computed using **fastText** (Δ(IAS, AST)). All the measures are for T_1. The hyper-parameter used for node2vec training - dimension of vectors (d) = 128.

Features used	Precision	Recall	F-measure
Δ(ED, SS, APL)	**0.86**	**0.56**	**0.68**
Δ(IAS, AST)	0.67	0.45	0.54
Δ(ED, SS, APL, IAS)	0.85	0.55	0.67
Δ(ED, SS, APL, AST)	0.84	0.52	0.64
Δ(ED, SS, APL, IAS, AST)	0.89	0.52	0.65

performance of simple network measures (ED, SS, APL) consistently for both T_1 and T_2.

All these analysis using network embeddings shows that even though network representation learning methods are supposed to capture network properties better by embedding the nodes in the network to a vector space, for a task like novel word sense detection explicit complex network features prove to be more useful, despite they are more efficient to compute. We see the same trend even when we try to obtain the embeddings directly from the corpus using fastText (described in Section 5.4), leading to the fact that a carefully curated small set of network measures can be more useful than high dimensional embeddings for this task of novel sense detection.

5.4. Evaluation of features obtained from fastText embeddings

In this analysis we investigate whether embeddings obtained directly from corpus using fastText (Bojanowski et al., 2017) can help boosting the performance for this task. The results for T_1 and T_2 are presented in Table 19 and Table 20, respectively. We see that using only IAS and AST does not help to improve the F-measure compared to using only the network measures, whereas when both these types of features are merged, it helps to boost the performance in different combinations for different time period pairs (T_1 and T_2).

5.5. Error analysis

We further analyze the cases, which are labeled as 'true birth' by the SVM but are evaluated as 'false' by the human evaluators. We find that in most of such cases, the sense cluster reported as 'birth' contained many new terms (and therefore, the network properties have undergone change) but the implied sense was already present in one of the previous clusters with *very few common words* (and therefore, the new cluster contained $> 80\%$ new words and is being reported as 'birth' in Mitra et al. (2014)). Two such examples are given in Table 21. The split-join algorithm proposed in Mitra et al. (2014) needs to be adapted for such cases.

We also analyze the 'true negatives' cases, which are labeled as 'false birth' by the SVM filtering but are evaluated as 'true' by the human evaluators. Two such examples are given in Table 22. By looking at the feature values of these cases, it is clear that the network structure of the induced subgraph is not changing much, yet they undergo sense change. The probable reason could be that the target word was not in the network of the induced subgraph in the old time point and enters into it in the new time point. Our SVM model is unable to detect this single node injection in a network so far. Handling these cases would be an immediate future step to improve the recall of the system.

We also observe that even though for time point T_1, we get good results for feature combination Δ(ED, SS, APL, AST) while using Deepwalk, it fails to improve the performance for T_2. Therefore, we dig into the reason behind the fall in the performance for T_2 and

Table 20
Comparison of performance of simple complex network measures (Δ(ED, SS, APL)) vs combination with features computed using **fastText** (Δ(IAS, AST)). All the measures are for T_2. The hyper-parameter used for node2vec training - dimension of vectors (d) = 128.

Features used	Precision	Recall	F-measure
Δ(ED, SS, APL)	0.74	0.61	0.67
Δ(IAS, AST)	0.86	0.52	0.65
Δ(ED, SS, APL, IAS)	0.74	0.61	0.67
Δ(ED, SS, APL, AST)	0.75	0.65	0.70
Δ(ED, SS, APL, IAS, AST)	**0.79**	**0.65**	**0.71**

Table 21
Example 'false positives' after SVM filtering (T_1). These words are flagged 'true birth' by SVM but manually evaluated as 'false'.

Word	'birth' cluster	Old cluster
aftercare	care, clinic, outpatient, ···	treatment, therapy, hospitalization, ···
electrophoresis	labeling, analysis, profiling, ···	analysis, counting, procedure, ···

Table 22
Example cases, labeled by SVM as 'false birth' but flagged as 'true birth' by annotators (T_1). The fractional change of the network measures is very low, leading to erroneous classification by SVM.

Word	'birth' cluster	Δ(ED, SS, APL)
baseplate	flywheel, cylinder, bearings, ···	0.06, −0.08, −0.84
grating	beam, signal, pulse, ···	0.2, −0.05, −0.88

do error analysis. Note that, the training set is from T_1 and while the mean of Δ(AST) for 'true birth' cases is 1.02, the value is 0.52 for 'false birth' cases. However, there are examples like 'tans', 'guitarist', 'conformist', etc., which are correctly predicted as 'false' by the classifier model using only complex network measures (Δ(ED, SS, APL)) as features, but are wrongly flagged as 'true' when we add Δ(AST) to the feature set. On analysis, we find that the Δ(AST) values for these cases are 1.93, 2.33 and 1.68, respectively. Similarly, target words like 'regularization', 'rewarming' are correctly predicted as 'true' by the classifier model using only the complex network measures (Δ(ED, SS, APL)) as features, but are wrongly flagged as 'false' when we add Δ(AST) to the feature set, because of the low values, 0.59 and 0.44, respectively. Network embedding frameworks attempt to put nodes with similar network properties close in the vector space and the properties include both the local and global neighborhood structures. As the training set is from T_1 and test set is from T_2, it seems that the network pattern, especially the global pattern which is captured by network embedding feature (AST), does not change in the same way for T_1 and T_2 leading to different range of feature values for both the classes in training and test set whereas the local neighborhood changing pattern captured by complex network measures are close for T_1 and T_2, leading to decent performance.

6. Detection of known shifts

So far, we have reported experiments on discovering novel senses from data and measured the accuracy of our method using manual evaluation. In this section, we evaluate the diachronic validity of our method on another task of detecting known shifts. We test whether our proposed method is able to capture the known historical shifts in meaning. For this purpose, we create a reference list L of **15** words that have been suggested by prior work (Eger & Mehler, 2016; Hamilton et al., 2016a; 2016b) as having undergone a linguistic change and emerging with a novel sense. Note that, we focus only on nouns that emerge with a novel sense between 1900 and 1990. The goal of this task is to find out the number of cases for which our method is able to detect a novel sense from the list L, which in turn would prove the robustness of our method.

Data: Consistent with the prior work, we use the Corpus of Historical American (COHA).[6] COHA corpus is carefully created to be genre balanced and is a well constructed prototype of American English over 200 years, from the time period 1810 to 2000. We extract the raw text data of two time slices: 1880–1900 and 1990–2000 for our experiment.

Experiment details and results: We first construct distributional thesauri (DT) networks (Riedl & Biemann, 2013) for the COHA corpus at two different time periods, 1880–1900 and 1990–2000. We apply Chinese Whispers algorithm (Biemann & Bosch, 2011) to produce a set of clusters for each target word in the DT network. The Chinese Whispers clusters for the target word 'web' are shown in Fig. 5. Note that we have reported only some of the representative words for each cluster. Each of the clusters represents a particular sense of the target. We now compare the sense clusters extracted across two different time periods to obtain the suitable signals of

[6] https://corpus.byu.edu/coha

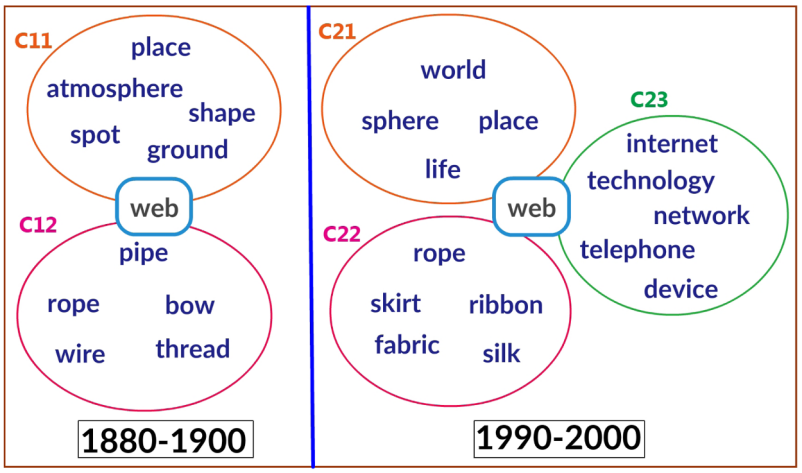


Fig. 5. Chinese Whisper clusters for the target word 'web' extracted from COHA corpus for the time periods 1880–1900 and 1990–2000.

Table 23
Example cases from the training set used for the experiment on detecting known shifts. Evaluation has been done by annotators.

Word	'birth' cluster	Manual Evaluation
caller	phone, message, operator, customer, ···	**Yes**, communication system related sense
courier	transport, purchase, company, delivery, ···	**Yes**, marketing related sense
public	student, economist, general, ···	**No**
richness	joy, happiness, stress, ···	**No**

Table 24
Example cases from COHA corpus, having linguistic shifts as suggested by prior literature and correctly detected by our approach. The discriminative feature shows the network measure which has changed the most over time.

Word	'birth' cluster	Discriminative feature
virus	weapon, system, aircraft ···	Δ(SSM)
cell	network, satellite, phone, ···	Δ(SSM)
monitor	computer, TV, screen, ···	Δ(ED)
axis	missile, fire, satellite, ···	Δ(ED)
broadcast	TV, cable, service, ···	Δ(APL)
check	wage, donation, fee, ···	Δ(APL)
film	show, concert, script, ···	Δ(ED)
focus	concern, ambition, ···	Δ(APL)
major	university, discipline, ···	Δ(APL)
program	project, database, testing, ···	Δ(ED)
record	tape, card, disc, copy ···	Δ(SSM)
web	Web, Internet, network ···	Δ(ED)

sense change following the approach proposed in Mitra et al. (2014). After getting the novel sense clusters, we pick up 50 random samples, of which 25 cases are flagged as 'true birth' while the other 25 cases are flagged as 'false birth' by manual evaluation. We use these 50 samples as our training set for classification using SVM. Some of the examples of this training set are presented in Table 23. We ensure that none of the words in the list L is present in the training set. Using this training set for our proposed SVM classifier, we are successfully able to detect **80%** of the cases (12 out of 15) from the list L as having a novel sense. Table 24 presents all of these detected words along with the novel senses and the discriminative network feature. Our method is unable to detect three cases -'gay', 'guy' and 'bush'. For 'gay', since there is no sense cluster in the older time period with 'gay' being a noun, cluster comparison does not even detect the 'birth' cluster of 'gay'. The 'birth' sense clusters for 'guy', 'bush' in the new time period, as detected by split-join algorithm contain general terms like "someone, anyone, men, woman, mother, son" and "cloud, air, sky, sunlight" respectively. As the network around these words did not change much over time, our method found it difficult to detect. Note that even though COHA corpus is substantially smaller than the Google n-grams corpus, our approach produces promising results, showing the usability of the proposed method with corpora of limited size as well.

7. Conclusion

In this study, we dealt with the task of improving the performance of novel sense detection task by borrowing concepts from complex network theory, which is an attempt of first of its kind. In order to improve the precision of detecting words evolved with a new sense over time, we demonstrated how the change in the network properties of the induced subgraphs from a sense cluster can be used. In addition, to investigate the superiority of complex network measures in this task, we explore the measures computed from the network representation learning framework as well. Manual evaluation over two different time period pairs shows that the proposed SVM classification approach boosts the precision values from 0.23–0.32 to 0.74–0.86 with a decent F-measure value of 0.67–0.68 when fed with only complex network measures. Even though the combination of complex network measures and network embedding measures improve the precision further to 0.76–0.91 in different scenarios, the F-measure falls significantly proving the superiority of complex network measures over network embedding measures. Note that, using only network embedding measures as features to the classification model leads to very poor performance compared to the model using only complex network measures. This study also shows that if we can intelligently apply complex network theory to come up with some intuitive measures to be used in a problem dealing with only local network structures, it can produce better or comparable performance than the network embedding measures which is otherwise computationally heavy to compute. Finally, from the experiments on the COHA corpus, we have also shown that our approach can reliably detect the words known to have sense shifts. The gold standard dataset prepared by us for validating novel sense detection is one of our main contributions and is made available publicly[7].

In future, we plan to apply our methodology to different genres of corpus, like social network data, several product or movie reviews data which are becoming an increasingly popular source for opinion tracking, to identify short-term changes in word senses or usages. These analyses would also provide insights into the evolution of language in a short span of time. Our ultimate goal is to prepare a generalized framework for accurate detection of sense change across languages and investigate the triggering factors behind language evolution as well.

CRediT authorship contribution statement

Abhik Jana: Conceptualization, Data curation, Writing - original draft, Investigation, Software, Validation, Writing - review & editing. **Animesh Mukherjee:** Conceptualization, Investigation, Supervision, Writing - review & editing. **Pawan Goyal:** Conceptualization, Investigation, Supervision, Writing - review & editing.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.ipm.2019.102173 .

References

Antiqueira, L., Nunes, M.d. G. V., Oliveira Jr, O., & Costa, L.d. F. (2007). Strong correlations between text quality and complex networks features. *Physica A: Statistical Mechanics and its Applications, 373*, 811–820.

Bamler, R., & Mandt, S. (2017). *Dynamic word embeddings. Proceedings of the 34th international conference on machine learning-volume 70*. JMLR. org380–389.

Baskaya, O., & Jurgens, D. (2016). Semi-supervised learning with induced word senses for state of the art word sense disambiguation. *JAIR, 55*, 1025–1058.

Biemann, C. (2006). *Chinese whispers: an efficient graph clustering algorithm and its application to natural language processing problems. Proceedings of the first workshop on graph based methods for natural language processing*. Association for Computational Linguistics73–80.

Biemann, C., & Bosch, A.v.d. (2011). *Structure discovery in natural language*. Springer Science & Business Media.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics, 5*, 135–146.

Bonneau, J., Just, M., & Matthews, G. (2010). *Whatâs in a name? Financial cryptography and data security*. Springer98–113.

Bordag, S. (2006). *Word sense induction: Triplet-based clustering and automatic evaluation. EACL*. Citeseer.

Cook, P., Lau, J. H., McCarthy, D., & Baldwin, T. (2014). *Novel word-sense identification. Coling*1624–1635.

Di Carlo, V., Bianchi, F., & Palmonari, M. (2019). Training temporal word embeddings with a compass. *Proceedings of the AAAI Conference on Artificial Intelligence, 33*(01), 6326–6334. https://doi.org/10.1609/aaai.v33i01.33016326.

Dorow, B., & Widdows, D. (2003). *Discovering corpus-specific word senses. Proceedings of EACL-volume 27*9–82.

Dorow, B., Widdows, D., Ling, K., Eckmann, J.-P., Sergi, D., & Moses, E. (2004). Using curvature and markov clustering in graphs for lexical acquisition and word sense discrimination. arXiv preprint cond-mat/0403693,.

Eger, S., & Mehler, A. (2016). *On the linearity of semantic change: Investigating meaning variation via dynamic graph models. ACL*52–58.

Erk, K. (2006). *Unknown word sense detection as outlier detection. Proceedings of the main conference on human language technology conference of the NAACL*128–135.

Erk, K., & Pado, S. (2007). *Towards a computational model of gradience in word sense. Proceedings of IWCS-7, Tilburg, The Netherlands*.

Ferrer i Cancho, R. (2004). R.; koehler, r.; solé, rv patterns in syntactic dependency networks. *Physical Review E, 69*, 32767.

Ferrer i Cancho, R., Capocci, A., & Caldarelli, G. (2007). Spectral methods cluster words of the same class in a syntactic dependency network. *International Journal of Bifurcation and Chaos, 17*(07), 2453–2463.

Ferrer i Cancho, R., & Solé, R. V. (2001). The small world of human language. *Proceedings of the Royal Society of London B: Biological Sciences, 268*(1482), 2261–2265.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin, 76*(5), 378.

Frermann, L., & Lapata, M. (2016). A bayesian model of diachronic meaning change. *Transactions of the Association for Computational Linguistics, 4*, 31–45.

Goldberg, Y., & Orwant, J. (2013). *A dataset of syntactic-ngrams over time from a very large corpus of english books. Second joint conference on lexical and computational semantics (* sem)1. Second joint conference on lexical and computational semantics (* sem)* 241–247.

Grover, A., & Leskovec, J. (2016). *node2vec: Scalable feature learning for networks. Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM855–864.

[7] https://tinyurl.com/u8rsbkq

Gulordava, K., & Baroni, M. (2011). *A distributional similarity approach to the detection of semantic change in the google books ngram corpus. Proceedings of the gems 2011 workshop on geometrical models of natural language semantics*67–71.
Hamilton, W. L., Leskovec, J., & Jurafsky, D. (Leskovec, Jurafsky, 2016a). *Cultural shift or linguistic drift? comparing two computational measures of semantic change. Proceedings of the 2016 conference on EMNLP*2116–2121.
Hamilton, W. L., Leskovec, J., & Jurafsky, D. (Leskovec, Jurafsky, 2016b). *Diachronic word embeddings reveal statistical laws of semantic change. ACL*1489–1501 Berlin, Germany
Jana, A., Mukherjee, A., & Goyal, P. (2019). *Detecting reliable novel word senses: a network-centric approach. Proceedings of the 34th ACM/SIGAPP symposium on applied computing.* ACM976–983.
Jatowt, A., & Duh, K. (2014). *A framework for analyzing semantic change of words across time. Proceedings of the 14th ACM/IEEE-SC joint conference on digital libraries*229–238.
Kenter, T., Wevers, M., Huijnen, P., & De Rijke, M. (2015). *Ad hoc monitoring of vocabulary shifts over time. Proceedings of the 24th ACM international on conference on information and knowledge management.* ACM1191–1200.
Kilgarriff, A., Rychly, P., Smrz, P., & Tugwell, D. (2004). Itri-04-08 the sketch engine. *Information Technology, 105*, 116.
Kulkarni, V., Al-Rfou, R., Perozzi, B., & Skiena, S. (2015). *Statistically significant detection of linguistic change. Proceedings of the 24th international conference on world wide web*625–635.
Kutuzov, A., Øvrelid, L., Szymanski, T., & Velldal, E. (2018). *Diachronic word embeddings and semantic shifts: a survey. Proceedings of the 27th international conference on computational linguistics*1384–1397.
Kwong, O. Y. (1998). Aligning wordnet with additional lexical resources. *Usage of WordNet in Natural Language Processing Systems*.
Lau, J. H., Cook, P., McCarthy, D., Gella, S., & Baldwin, T. (2014). *Learning word sense distributions, detecting unattested senses and identifying novel senses using topic models. Proceedings of ACL*259–270.
Lau, J. H., Cook, P., McCarthy, D., Newman, D., & Baldwin, T. (2012). *Word sense induction for novel sense detection. Proceedings of EACL*591–601.
Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development, 2*(2), 159–165.
Mihalcea, R., & Moldovan, D. I. (1999). *An automatic method for generating sense tagged corpora. AAAI/IAAI*461–466.
Mihalcea, R., & Nastase, V. (2012). *Word epoch disambiguation: Finding how words change over time. Proceedings of the 50th annual meeting of the association for computational linguistics: Short papers-volume 2*259–263.
Mitra, S., Mitra, R., Riedl, M., Biemann, C., Mukherjee, A., & Goyal, P. (2014). *That's sick dude!: Automatic identification of word sense change across different timescales. ASL*1020–1029.
Pantel, P., & Lin, D. (2002). *Discovering word senses from text. Proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining*613–619.
Pardo, T. A. S., Antiqueira, L., Nunes, M.d. G. V., Oliveira Jr, O. N., & da Fontoura Costa, L. (2006). *Using complex networks for language processing: The case of summary evaluation. Communications, circuits and systems proceedings, 2006 international conference on4. Communications, circuits and systems proceedings, 2006 international conference on* 2678–2682.
Pedersen, T., & Bruce, R. (1998). *Knowledge lean word-sense disambiguation. AAAI/IAAI*800–805.
Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). *Deepwalk: Online learning of social representations. Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining.* ACM701–710.
Ramiro, C., Srinivasan, M., Malt, B. C., & Xu, Y. (2018). Algorithms in the historical emergence of word senses. *Proceedings of the National Academy of Sciences, 115*(10), 2323–2328.
Raviv, A., Markovitch, S., & Maneas, S.-E. (2012). *Concept-based approach to word-sense disambiguation. AAAI.*
Riedl, M., & Biemann, C. (2013). *Scaling to large3 data: An efficient and effective method to compute distributional thesauri. EMNLP*884–890.
Rudolph, M., & Blei, D. (2018). *Dynamic embeddings for language evolution. Proceedings of the 2018 world wide web conference on world wide web*1003–1011.
Sahlgren, M. (2002). *Towards a flexible model of word meaning. AAAI spring symposium*25–27.
Szymanski, T. (2017). *Temporal word analogies: Identifying lexical replacement with diachronic word embeddings. Proceedings of the 55th annual meeting of the association for computational linguistics (volume 2: Short papers)2. Proceedings of the 55th annual meeting of the association for computational linguistics (volume 2: Short papers)* 448–453.
Tahmasebi, N., Borin, L., & Jatowt, A. (2018). Survey of computational approaches to diachronic conceptual change detection. *Computational Linguistics, 1*(1).
Tahmasebi, N., Risse, T., & Dietze, S. (2011). *Towards automatic language evolution tracking, a study on word sense tracking. Workshop on knowledge evolution and ontology dynamics (EVODYN 2011), co-located with ISWC 2011*.
Tang, X. (2018). A state-of-the-art of semantic change computation. *Natural Language Engineering, 24*(5).
Teh, Y. W., Jordan, M. I., Beal, M. J., & Blei, D. M. (2006). Hierarchical dirichlet processes. *Journal of the American Statistical Association, 101*(476).
Turnu, I., Marchesi, M., & Tonelli, R. (2012). *Entropy of the degree distribution and object-oriented software quality. Proceedings of the 3rd international workshop on emerging trends in software metrics*77–82.
Viera, A. J., Garrett, J. M., et al. (2005). Understanding interobserver agreement: The kappa statistic. *Fam Med, 37*(5), 360–363.
Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications. 8*. Cambridge University Press.
Wu, Z., & Giles, C. L. (2015). *Sense-aaware semantic analysis: A multi-prototype word representation model using wikipedia. Twenty-ninth AAAI conference on artificial intelligence*.
Yao, Z., Sun, Y., Ding, W., Rao, N., & Xiong, H. (2018). *Dynamic word embeddings for evolving semantic discovery. Proceedings of the eleventh ACM international conference on web search and data mining*673–681.