Towards Understanding Word Embeddings: Automatically Explaining Similarity of Terms

Yating Zhang
Kyoto University
Yoshida-honmachi, Kyoto, Japan
zhang@dl.kuis.kyoto-u.ac.jp

Adam Jatowt
Kyoto University
Yoshida-honmachi, Kyoto, Japan
adam@dl.kuis.kyoto-u.ac.jp

Katsumi Tanaka
Kyoto University
Yoshida-honmachi, Kyoto, Japan
tanaka@dl.kuis.kyoto-u.ac.jp

***Abstract*—Word embedding techniques (e.g., *Word2Vec*, *GloVe*) have been recently used for variety of applications with quite good rate of success. They allow to capture word semantics and syntactics with decreased dimensionality based on the concept of distributional vector representations. Vector representations can be then used for similarity comparison. However, if we treat the word embeddings as a kind of encryption process, it is difficult to decrypt their meaning. This makes it problematic to justify why particular terms should be considered similar as well as to prove that the overall quality of the trained vector space is high. Evaluating the accuracy of the similarity computation between any two given terms is difficult due to the lack of concrete evidences to explain and support the similarity. In this paper, we propose a novel way to automatically extract evidences represented as term pairs to explain the similarity of arbitrary terms. Our approach is unsupervised and can be applied to either homogeneous or heterogeneous vector spaces.**

***Index Terms*—word embeddings; similarity explanation; comparison criteria; analogy; search**

I. Introduction

Word embedding techniques have been frequently used for variety of applications with remarkably high level of success. They have been especially popularized by Mikolov et al. [1] who introduced the Skip-gram model - an efficient method based on a simplified Neural Network model for learning high-quality vector representations of words in large amount of text data. GloVe is another popular word embedding technique proposed by Pennington et al. [2], in which the word embeddings are trained by a global log bilinear regression model. Although different techniques are used in these models, both allow capturing and reflecting word semantics and syntactics. This naturally enables to perform word similarity comparison. Commonly, the similarity between two words is measured by calculating the cosine similarity between their vector representations. Embedded word representations as claimed by Mikolov et al. [1] and Pennington et al. [2] demonstrate linguistic regularities and patterns which can be expressed as linear operations in a vector space, such as vec(“Madrid”)−vec(“Spain”)≈vec(“Paris”)−vec(“France”) or vec(“man”)−vec(“woman”)≈vec(“king’)−vec(“queen”).

However, the word embedding technique is often treated as a “black box” process, in which it is difficult to “decrypt” the meaning behind the word embeddings to support explanation of the results of similarity calculation. This problem is particularly important when a user does not only want to know *how much* two words are similar to each other, but rather she wishes to know *why* they are similar. Simply outputting a similarity score is not convincing and does not answer the “why similar” question.

Furthermore, generating persuasive, explanatory term pairs to indicate relation and similarity (e.g., a term pair `Spain` and `France` for the input *Madrid* and *Paris*) could help to convince users that the quality of the word embedding training is sufficient. Note that depending on the type of the underlying data (e.g., news article corpora, tweet datasets or Wikipedia) and the size of such data (e.g., thousands or rather millions of documents) the results can be of varying quality [1], [3].

Automatically explaining the similarity of arbitrary terms is however not trivial. While the example of “Madrid” and “Paris” may be considered relatively easy as the compared terms share many common concepts which could be extracted from external resources such as Wordnet (e.g., both are `capitals` and both are located in `Europe`), the comparison between two analogical terms such as *iPod* and *Walkman* is not straightforward. This is because both the input words share very few identical context terms [4] (mainly, because they were products popular at different times: *iPod* in 2010s and *Walkman* in 1980s). *How can we explain the similarity of such terms and provide concrete evidence why the terms are positioned close to each other in the vector space?* Ideally, the output should indicate their non-trivial common aspects such as that both are *portable*, both utilize some storage media to store songs (`MP3` for *iPod* and `cassette` for *Walkman*), both were introduced by a single, dominant company (`Apple` for *iPod* and `Sony` for *Walkman*), and so on.

Note however that even in the case of presumably easy queries that tend to share many common context terms, the problem is difficult due to the variety of shared aspects of queried terms. For example, when trying to explain the similarity between *Madrid* and *Paris*, there will be multiple candidates of commonalities (e.g., both the cities have `trees` and `cars`). Obviously, these commonalities are not convincing enough to support term similarity judgment as they are not the key, shared aspects representative and distinctive enough for conducting the comparison. In other words, the `trees` and `cars` cannot indicate the actual reasons for similarity of *Madrid* and *Paris*. Therefore, special criteria are needed in order to provide convincing explanation of term similarity.

The last but not the least, the proposed approaches should be unsupervised without the need for manually selecting or defining the facets or characteristics needed for comparison. It is virtually impossible to prepare the set of rules or fixed aspects to perform comparison of arbitrary two terms (e.g., terms such as objects, persons, locations). Furthermore, since the word embeddings are trained using a particular document collection, different collections will lead to different resulting word embeddings. Therefore, the evidences for similarity explanation between compared terms should be also corpus specific to reflect the true reasons (learned from the very same dataset) as for why the terms are "close" in the vector space.

To tackle the above-discussed challenges, we introduce and define the novel research problem of automatically explaining term similarity. Particularly, we define the concept of "similarity" by the combination of "commonality" and "aligned difference" based on the nature of human's comparison processes explained in psychology science [5]. We then propose several criteria to ensure the quality of the retrieved evidences: *relevance*, *semantic similarity*, *relational similarity* and *systematicity*. Our methods return in a fully unsupervised fashion the set of "evidences" indicating similar aspects of the given two terms such that the extracted evidences are convincing, relevant and important for the similarity judgment.

While our solution is generic and can be applied to many scenarios, in the current implementation, we focus on the specific case of *explaining across-time similarity of terms*. That is, we compare entities from different time periods (using non-overlapping datasets of documents published at these periods) and explain their analogy. Explaining the time-based analogy is especially useful because of the lack of dictionaries, lexical bases and thesauri that would highlight similar points in entities dispersed over time such as one entity from the present and the other from some period in the past.

The methods that we propose can have diverse applications. First, as stated before, they can be used for explaining term similarity and for investigating the quality of word embeddings. Especially, they can be used for answering analogy like questions (a :? :: c :?) that cannot be easily answered by current QA systems [6], [7]. Second, the proposed solutions can assist professionals from various fields in their work, for example, in *comparative historical studies* [8]. In particular, the proposed techniques could form components of advanced interfaces for interacting with document archives and digital libraries [9], [10]. Finally, semantic databases such as Yago [11] or DBpedia [12] could be enriched by adding novel kind of "similarity" links with their corresponding explanations.

To sum up, our contributions are as follows:

1) We introduce a novel problem of explaining similarity of terms in semantic vector spaces.
2) We define several criteria to extract effective evidences to support similarity justification between any two terms.
3) We propose two approaches for selecting explanatory terms based on the local and global characteristics.
4) We evaluate the proposed approaches on the New York Times Archive [13] and demonstrate that the detected evidences can effectively explain the similarity between terms such that the terms coming from different times.

II. Related Work

Analogical relation detection [14], [15], [16] is related to our work. Structure Mapping Engine (SME) [14] was the original implementation of Structure Mapping Theory (SMT) [17] that explains how humans reason with analogy. Later, Turney proposed Latent Relational Mapping Engine (LRME) [15] that extracts lexical patterns in which words co-occur to measure relational similarity. The problem of explaining the analogy was however never explicitly approached. One reason for this could be that creating or recognizing analogy is already quite difficult. Also, explaining the similarity of quite similar terms (or very common terms) might have been considered as a simple task for humans that does not require automatic computation. Note however that in the case of entity comparison in heterogeneous spaces we cannot assume that users know commonalities and aligned differences of entities at different domains (e.g., different times), especially, ones from very different spaces (e.g., the present time period and some period in the distant past). Neither, we can assume that ready knowledge bases or dictionaries exist for past domains. Thus, the similarity explanation task is clearly needed.

Our work is also related to the task of proportional analogy extraction. Typically, this task is defined as follows. Given any three items in the equation $a : b :: c : d$, the problem is to extract the fourth one such that the relation between a (e.g., Apple) and b (e.g., iPad) is proportionally analogical to the relation between c (e.g., Microsoft) and d (e.g., Surface). Different types of models have been proposed for this task [15], [18], [19], [20], [21]. They all follow a process of first detecting the relation/lexical patterns between two items in one side (e.g., the lexical pattern between Apple and iPad which is "the tablet of"), and then of searching for the corresponding entity d such that c and d share the same relation (e.g., search for the tablet produced by Microsoft; Surface being the result). The biggest difference from our work and theirs is that our input is two items a and c. We then detect b and d to make the equation $a : b :: c : d$ hold in order to explain the similarity/analogy between a and c. In this case, the relation between a and b (or c and d) is obviously unknown which makes the task much harder. Furthermore, b and d are of the special kind since they are explanatory, similar aspects of a and c, respectively. Another major difference is that most of the prior works [15], [18], [19], [21] highly rely on the existing relational graphs or lexical databases (e.g., WordNet, Yago, Wikipedia). However our work generates relational graphs entirely based on the underlying corpus without using any external sources, so our methods are flexible in detecting analogical relations occurring in any corpus the user is interested in.

III. Background and Problem Definition

The nature of similarity has been widely studied in psychology and cognitive science [17], [5], [22]. The main claim is that the similarity is like analogy, and computing similarity involves not only the comparison of objects' attributes but also the alignment of relational structures constructed from these attributes [5]. In this paper, we define the similarity between two terms as the union of their *commonalities* and *aligned differences* based on the theory of *Structural Alignment Model* [5].

Commonality is defined as a pair of identical features of two entities. For example, `music` :: `music` is a commonality of *iPod* and *Walkman* as both are designed to play music.

Aligned difference is defined as a pair of features which have the same relation to both the entities but have different values. For instance, `MP3` :: `cassette` is an aligned difference of *iPod* and *Walkman*. Although these two features are different in their literal forms, they share the similar relation of being the storage media for their corresponding entities.

Based on these definitions, we cast the problem of explaining the across-time similarity as below:

PROBLEM STATEMENT. Given two entities, e^A and e^B, the task is to find the set of their commonalities and aligned differences, $S_{sim}(e^A,e^B) = \{w_1^A \approx w_1^B,\ w_2^A \approx w_2^B,\ \ldots,\ w_l^A \approx w_l^B\}$, where w_i^A and w_i^B are terms related, respectively, to e^A and e^B. Note that the entities can be from the same domain or from different domains. The latter case can be regarded as a generlized version of the former. A domain is loosely defined here to be the collection of domain-related documents (e.g., research papers, documents about a particular country, documents created at a particular time period). In the experiments, we focus on the particular case of different domains - the datasets of documents published at different time periods.

To sum up, in this work, we model the task as a set construction problem, where the member of the set is a term pair denoting either commonalities or aligned differences of input entities. The constituent terms of each pair are selected from the contexts of the entities.

IV. Quality-based Similarity Detection

As explained in Sec. III, the task is to select a set of term pairs denoting similarities between input entities. In this section, we first discuss the desired criteria of terms useful for indicating the similarity of entities. We then propose several measures to select high quality pairs based on these criteria.

A. Criteria for Selecting Term Pairs

Let $\langle w_i^A, w_i^B \rangle$ denote a pair of terms where w_i^A appears in the context of entity e^A and w_i^B occurs in the context of e^B. A high quality term pair has the following characteristics: (a) *Relevance*. Intuitively, the ideal terms should be distinctive and typical for queried entities to represent their characteristic attributes. In the example of *iPod* and *Walkman*, both the entities may have the same color (e.g., `blue`), yet the color attribute cannot distinguish them from other products and is not specific to these objects. Formally speaking, both w_i^A and w_i^B are required to be highly related to their corresponding entities (e.g., `portable` to *iPod* (e^A) and `portable` to *Walkman* (e^B)). (b) *Semantic Similarity*. It is commonly expected that if two entities are similar then usually their attributes should be similar, too. Thus, in order to explain their similarities, it is necessary to list up their similar attributes. For example, `MP3` (w_i^A) and `cassette` (w_i^B) indicate the similar concept of storage media for the entities *iPod* and *Walkman*, respectively. (c) *Relational Similarity.* We use relational similarity to refer to the structural consistency between two similar things. The relation between w_i^A and e^A should be similar to the one between w_i^B and e^B (e.g., `MP3` is the format used by *iPod*, and `cassete` is the format used by *Walkman*). We explain how to compute each of the above-discussed criteria in the next section.

B. Term Pair Quality Estimation

Relevance. The relevance of $\langle w_i^A, w_i^B \rangle$ to the query entities is measured using a variant of Pointwise Mutual Information (PMI). We first calculate the strength of PMI between w_i^A and e^A, and that of w_i^B and e^B. We then multiply them to estimate the combined relevance of the term pair to both the entities.

$$\begin{aligned} Rel\langle w_i^A, w_i^B \rangle &= pmi(w_i^A, e^A) \cdot pmi(w_i^B, e^B) \\ &= \log \frac{p(w_i^A, e^A)}{p(w_i^A)p(e^A)} \times \log \frac{p(w_i^B, e^B)}{p(w_i^B)p(e^B)} \end{aligned} \tag{1}$$

Semantic Similarity. We measure semantic similarity between the two context terms constituting a pair $\langle w_i^A, w_i^B \rangle$. As discussed before, we allow for the case in which the two input terms come from heterogeneous vector spaces (each term is represented in different domains such as different time period). We then use a generic computation technique to measure the semantic similarity between two terms from diverse domains. It is estimated by calculating the cosine similarity between the transformed representation of one term (e.g., w_i^A) represented as $\mathbf{M} \cdot \mathbf{w_i^A}$ and the representation of a term (w_i^B) from the other vector space. The idea is to utilize a transformation matrix $\mathbf{M}$ to map the terms from one space into the other semantic space in order to make them comparable in the same target vector space. For this, we apply the transformation technique which will be discussed in Sec. VII-B. Note that if the two compared terms come from the same vector space (e.g., they are represented using the same domain such as the same time period), then $\mathbf{M}$ is an identity matrix. $\mathbf{w_i^A}$ and $\mathbf{w_i^B}$ are column vectors.

$$S_{sim}\langle w_i^A, w_i^B \rangle = cos(\mathbf{M} \cdot \mathbf{w_i^A}, \mathbf{w_i^B}) \tag{2}$$

Relational Similarity. In contrast to the Semantic Similarity, the Relational Similarity estimates whether the relation between w_i^A and e^A is aligned with the one between w_i^B and e^B. To represent the relation, we follow the linear operations typically supported by word embedding techniques. For example, for capturing the relation between w_i^A and e^A, we take the difference of their vector representations, $\mathbf{w_i^A} - \mathbf{e^A}$ (see Eq. 3). Such linear analogical reasoning is also suggested in [3], [2]. Similar to the computation of the Semantic Similarity, the transformation matrix $\mathbf{M}$ is used here for coping with heterogeneous vector spaces.

$$R_{sim}\langle w_i^A, w_i^B \rangle = cos(\mathbf{M} \cdot (\mathbf{w_i^A} - \mathbf{e^A}), (\mathbf{w_i^B} - \mathbf{e^B})) \tag{3}$$

Term Pair Quality. After determining each criteria, a direct way is to compute the quality score of a candidate term pair by aggregating the three above-discussed measures. The higher the score is, the higher the quality of the term pair to explain the similarity between the given two entities.

$$Q\langle w_i^A, w_i^B\rangle = Rel\langle w_i^A, w_i^B\rangle \cdot S_{sim}\langle w_i^A, w_i^B\rangle \cdot R_{sim}\langle w_i^A, w_i^B\rangle \quad (4)$$

V. Systematicity-based Similarity Detection

Sec. IV introduced a direct way to measure the quality score of a term pair; that is, the score calculation of a pair was independent of computing the scores of other pairs. However, the relations (or dependencies) between different term pairs can actually provide additional signal useful for choosing good explanatory pairs. Our reasoning is based on the notion of systematicity used in psychology - a central factor controlling what information humans consider when comparing objects [5]. According to the *Systematicity Principle* when two compared entities have multiple features, the pair of the features that preserves the maximal connected relational structure is preferred for similarity comparison. In other words, a term pair belonging to a bigger structure is preferred over isolated pair or pair in smaller structure. Systematicity can be regarded as coherence reflecting term pairs' dependency on other pairs.

We propose to adopt the systematicity idea to our objective based on the following hypothesis:

A term pair is a good pair if it aligns well with many other good term pairs.

In other words, we try to find the set of terms which can reflect the largest structural alignment between the attributes of two entities. In such a sense, two requirements have to be satisfied for selected term pairs: (1) the quality of each term pair should be high (such that the term pair alone is already reasonably good for explaining the similarity between the compared entities) (2) the term pair should be supported by many good pairs in order to achieve the structural alignment. Alignment means here the relational correspondence of term pairs. Based on these two points, we propose a graph-based approach which is conceptually portrayed in Fig. 1. Let $G = (\Pi, E)$ be a graph composed of the set of term pairs, Π, used as vertices and the set of connecting them edges, E. The connection strength (weight) between any two nodes (two term pairs), π_i ($\langle w_i^A, w_i^B\rangle$) and π_j ($\langle w_j^A, w_j^B\rangle$) is estimated based on:

1) **Node pair quality**: $qual(\pi_i, \pi_j)$ - the degree to which the two nodes have high quality.
2) **Node-to-node relational alignment**: $align(\pi_i, \pi_j)$ - the degree to which the two nodes are relationally aligned. It measures how much the relation between w_i^A and w_j^A is similar to the relation between w_i^B and w_j^B.

The scores of $qual$ and $align$ are measured by Eq. 5 and Eq. 6, respectively.

$$qual(\pi_i, \pi_j) = Q\langle w_i^A, w_i^B\rangle \cdot Q\langle w_j^A, w_j^B\rangle \quad (5)$$

$$align(\pi_i, \pi_j) = cos(\mathbf{M} \cdot (\mathbf{w_i^A} - \mathbf{w_j^A}), (\mathbf{w_i^B} - \mathbf{w_j^B})) \quad (6)$$

The weight of an edge (ψ_{ij}) between two nodes (term pairs) is then estimated as the aggregation of node pair quality and node-to-node relational alignment.

$$\psi_{ij} = qual(\pi_i, \pi_j) \cdot align(\pi_i, \pi_j) \quad (7)$$

Finally, we compute the systematicity score of a term pair reflecting the previously mentioned hypothesis. Specifically, the scores are calculated based on the random-walk algorithm as in Eq. 8[1]:

$$SQ(\pi_i) = (1-d) + d \cdot \sum_{\pi_j \in N(\pi_i)} \frac{\psi_{ji}}{\sum\limits_{\pi_k \in N(\pi_j)} \psi_{jk}} \cdot SQ(\pi_j) \quad (8)$$

where $N(\pi_i)$ denotes the neighbors of π_i and d is a damping factor set to 0.85. The systematicity score computation is similar to the calculation of TextRank algorithm [23]. However, in our approach a node is actually a term pair (instead of a term) and the edge weight depends on the alignment of term pairs incident with the edge and on their quality (see Fig. 1).

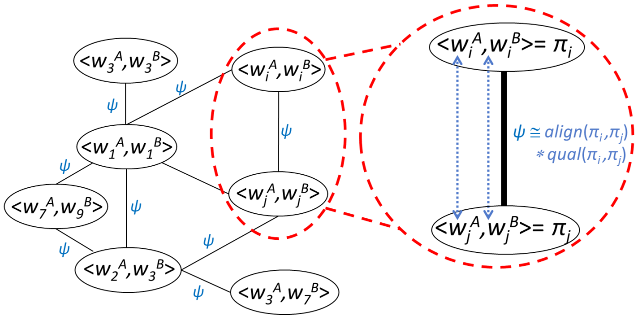

Fig. 1: Conceptual view of graph used for the systematicity-based similarity detection.

VI. Additional Processing

A. Result Diversification

A good set of pairs is the one in which the pairs are not only of high quality but are also different from each other. In other words, pair-level diversity within the returned set should be also considered. We adopt the concept of Maximal Marginal Relevance (MMR) [24] such that pairs are selected based on their quality as well as based on their dissimilarity to the already selected pairs.

$$S^* = \operatorname*{argmax}_{\pi_u \in U \setminus S} \{\lambda \cdot L(\pi_u) - (1-\lambda) \max_{\pi_i \in S} sim(\pi_u, \pi_i)\} \quad (9)$$

U is the ranked list of term pairs retrieved by either the quality-based similarity detection method (Eq. 4) or by the systematicity-based similarity detection (Eq. 8). S is the subset of U denoting the already selected pairs. and $U - S$ is their difference, i.e., the set of unselected word pairs in U. π_i and π_u denote the word pairs, $\langle w_i^A, w_i^B\rangle$ and $\langle w_u^A, w_u^B\rangle$, in the selected subset and unselected subset, respectively. $L(\pi_u)$ is the score of a term pair π_u computed by either Eq. 4 or by Eq. 8. Finally, $sim(\pi_u, \pi_i)$ denotes the semantic similarity

[1] We first pruned the graph by removing nodes with low Semantic Similarity scores to make the computation manageable. The convergence criterion was set to 1e-06.

between the two term pairs calculated by multiplying the cosine similarity between w_u^A and w_i^A with the one between w_u^B and w_i^B. Note that although the diversity is important, it is not the key technique in this research, thus to remove its impact on the evaluation, we apply the diversification procedure to our methods and to all the baselines used in experiments.

B. Extraction of Supporting Sentences

Sometimes users may not be able to understand similarity of entities based on generated terms. We then also provide supporting sentences as an additional information for setting the selected pairs in their contexts and for making them easier to be evaluated. For each retrieved term pair $\langle w_i^A, w_i^B \rangle$, we extract two representative sentences: one containing the entity e^A and term w_i^A, the other containing the entity e^B and term w_i^B. The objective of these two sentences is to implicitly address the relation between the entities and the terms constituting the selected pair $\langle w_i^A, w_i^B \rangle$. To discover the representative sentence for w_i^A, we first construct the set (denoted as SEN^A) of the sentences which contain both e^A and w_i^A. Then we extract terms that frequently occur within these sentences to form a feature vector weighted by term frequency. Next, we compare each sentence sen ($sen \in SEN^A$) with this feature vector selecting the one with the highest similarity (highest cosine similarity score). The same process is applied to extract sentences containing e^B and w_i^B. We note that for simplicity, no sentence is returned if the document collection does not contain any sentence with both the entity and a term belonging to the selected term pair.

VII. Experimental Setup

A. Datasets

As mentioned above, our approach allows explaining similarity between terms belonging to heterogeneous spaces. We test our approaches on a particular case of different vector spaces: document collections from different time periods. In particular, we use the New York Times Annotated Corpus [13], which has been frequently utilized in related studies [25], [4], [26]. It contains over 1.8 million newspaper articles published from 1987 to 2007. We choose the two time periods, [2002, 2007] and [1987, 1991], which are separated by sufficiently long gap. We then select query entities (e^A, e^B) such that e^A is from [2002, 2007] and e^B comes from [1987, 1991].

B. Constructing Transformation Matrix

Our goal is to compare entities from two disjoint time periods. However, the meaning of the same terms within the contexts of the compared entities may not be the same as the entities are represented in diverse time periods. Indeed, it is well-known that words change their semantics over time [27], [28]. We thus cannot directly match terms across time, i.e., by the equality of their literal forms. Even if we find the same term in both the time periods, there is no assurance that the term still denotes the same concept. On the other hand, sometimes different terms in different time periods may represent the same or very similar concept. Thus, we apply the transformation matrix $\mathbf{M}$ mentioned above to align two vector spaces for enabling similarity comparison.

Our approach is as follows. First, we separately train word embeddings to represent the vocabularies in each time period by utilizing Word2Vec [1], [3]. Then, we apply the transformation technique proposed in [26] to align the vocabularies of the two time periods, so that the words from one vector space can be compared to the ones in another space after transformation. The idea is to utilize semantically stable terms across time as anchors to bridge the two vector spaces. Once the mapping is found using the anchors, the other terms within the two spaces can be aligned by the similarity of their positions relative to the anchor terms in their own spaces. We choose as anchor terms the terms which have the same literal forms and which are sufficiently frequent in both the time periods (e.g., sky, river, man). One reason to choose the frequent terms as anchors is because they tend to be strongly "connected" (co-occurring) with many other terms. The other reason is that frequent terms are subject to relatively small semantic drift over time. As observed in several languages including English [29], [30], the more frequent a word is, the harder is to change its dominant meaning across time (or the longer time it takes for the meaning shift to occur).

Suppose there are k pairs of anchor terms $\{(x_1^A, x_1^B),\ldots,(x_k^A, x_k^B)\}$ where x_i^A is an anchor in one space (e.g.„ present time period) and x_i^B is its counterpart, that is, the same anchor in the other space (e.g., past time). The transformation matrix $\mathbf{M}$ is found by minimizing the differences between $\mathbf{M} \cdot \mathbf{x_i^A}$ and $\mathbf{x_i^B}$ (see Eq. 10). This is done by minimizing the sum of Euclidean 2-norms between the transformed query vectors and their counterparts. Eq. 10 is used for solving the regularized least squares problem ($\gamma = .02$) with regularization component used for preventing overfitting:

$$\mathbf{M} = \underset{\mathbf{M}}{\operatorname{argmin}} \sum_{i=1}^{k} \left\| \mathbf{M} \cdot \mathbf{x_i^A} - \mathbf{x_i^B} \right\|_2^2 + \gamma \left\| \mathbf{M} \right\|_2^2 \qquad (10)$$

k denotes here the size of anchor term set containing the top 5% frequent words in the intersection of vocabularies of the two time periods. This number has been experimentally found to perform best in aligning the tested two time periods [26].

Note that our approach is generic. For other types of heterogeneous spaces we need to train a dedicated transformation matrix to align the two vector spaces. The anchor terms in such a case should be terms having the same meaning in both the different spaces (e.g., frequent terms judged to represent the same concepts). On the other hand, if the similarity explanation task is conducted over the same vector space, then there is no need for transformation.

C. Test Sets

Since our problem is novel, there are no benchmark datasets available. We then manually created test sets containing entity pairs[2]. We chose entities that are potentially similar to each

[2] Test queries' listing is available at http://tinyurl.com/hbcdr3x.

other and which belong at equal rate to three types: objects including events (e.g., *iPod* vs. *Walkman*, *Iraq War* vs. *Gulf War*), persons (e.g., *Vladimir Putin* vs. *Boris Yeltsin*) and locations (e.g., *Germany* vs. *East Germany*). To consider diverse scenarios, half of the queries contain the same entity in both the past and present time periods (e.g., *Arnold Schwarzenegger*[2002,2007] vs. *Arnold Schwarzenegger*[1987,1991]), and half contains different entities. In total, we had 60 different query pairs covering 90 unique entities.

We have evaluated the total of $4,055$ term pairs for the experiments. In particular, we have leveraged the pooling technique [31] by pulling the top 20 retrieval results from 5 different systems (proposed methods and baselines). Three annotators judged every result (term pair) in the pool as for whether it indicates similarity of queried entities, producing, in total, $12,165$ judgments. The annotators did not know which systems generated which pairs as all the term pairs from the pool were alphabetically ordered for each query. They were encouraged to use external sources including Wikipedia and search engines in order to verify the quality of each result. The annotators could also see supportive sentences provided for every returned term pair (see Tab. V for examples). Each annotator took on average 70 hours[3] for completing the annotation task due to the need for studying the history and searching for details of each entity. A term pair was regarded as a correct answer, if at least two annotators have accepted it. The average Fleiss' Kappa [32] is 0.71, indicating *substantial agreement* across the raters (values above 0.61 are considered as substantial agreement [33]). The average rate of commonalities to aligned differences in the ground truth is 46%:54% (43%:57%, 50%:50% and 47%:53% for objects, persons and locations, respectively).

D. Evaluation Measures and Tested Methods

We use *Precision*, *Recall* and F_1 as metrics. *Precision* is computed as the ratio of correct term pairs within the top 20 returned results. *Recall* is calculated as the ratio of correct returned term pairs to the total number of the correct pairs.

For each tested method, we set up the same pre-processing and post-processing steps. In particular, for a given query (e^A, e^B), we extract the top relevant context terms of e^A, $\{w_1^A, ..., w_n^A\}$, and the top relevant context terms of e^B, $\{w_1^B, ..., w_n^B\}$ ($n = 500$) to be used for pair construction. We also apply the diversification procedure (see Sec. VI-A) to the results of all the methods (λ equals 0.1).

Baselines. We prepare three baselines:

(1) ***Overlap approach*** (**Overlap**): this method simply selects identical terms from the most relevant context terms of entities. Then the term pairs are ranked by the relevance score calculated using Eq. 1. The top 20 term pairs with the highest score are returned as results. **Overlap** approach has the advantage of being simple and fast. However, it only considers commonalities between entities ignoring their aligned differences. We use it to test whether the commonalities alone are enough for the similarity comparison.

(2) ***Adjusted Vector Space Model*** (**VSM**): Our work is related to the typical task of proportional analogy which requires solving the problem of $a : b :: c :?$. Yet, we note that our task is more difficult since we are actually solving $a :? :: c :?$, and we are the first to do so. The Vector Space Model (VSM) based method proposed by Turney et al. [34] is commonly adopted in solving the task of proportional analogy. Although the input to our methods is different from the one of Turney et al. [34] (two terms instead of three terms), we have adjusted their approach to fit to our problem. The fundamental idea of Turney's work is (1) to use the vector of terms ($\mathbf{s_i}$) to represent the semantic meaning of a term (w_i) and (2) to use a vector of terms/lexical patterns ($\mathbf{r_{ij}}$) to represent the relation between two terms (e.g., w_i and w_j). (3) Then, the semantic similarity between the terms and relational similarity between the term pairs can be measured by the cosine similarity between the corresponding vectors.

For creating the baseline, we represent the semantic meaning of a term by the TF-IDF weighted vector of its co-occurring terms. We then represent the relation between the entity term (e^A or e^B) and the context term (w_i^A or w_i^B) by the vector of terms co-occurring with both the entity term and the context term within the same sentences. Then, we rank the candidate term pairs by aggregating their semantic similarity and their relational similarity to corresponding entities, and we finally output the top 20 terms pairs as the results.

Note that the differences of our proposed baseline, called **VSM**, and the method in [34] is that we do not apply the limited set (128 in [34]) of lexical rules and patterns (e.g., ”X of Y”, ”X to Y”) to explicitly capture relations, but, instead, we use the aggregate of context terms which co-occur with both X and Y. Another difference is that we do not rely on a search engine to extract relationships as done in [34], but we directly utilize the underlying target corpus.

(3) ***Quality-based Similarity Detection without transformation*** (**QSD-NT**): the third baseline is set to be different from our proposed method **QSD** by removing the transformation process. It merges the datasets from the two time periods and trains one vector space over the combined dataset. This baseline is used to test the necessity of transformation when the entity comparison is conducted over heterogeneous spaces.

Proposed Methods. We test the two proposed methods: the **Quality-based Similarity Detection** (**QSD**) method (see Sec. IV) and the **Systematicity-based Similarity Detection** (**SSD**) method (see Sec. V) which leverages the graph-based infrastructure among the term pairs.

VIII. Experimental Results

The average scores for each method are shown in Tab. I. Tab. II presents several results for example queries. More results are also shown at the end of the paper (see Tab. VI). The main finding is that both our methods statistically significantly outperform the baselines by all the measures. In the following subsections we discuss the results in detail.

[3] On average, over 1h was taken to evaluate the pooled results for one query.

TABLE I: Main results. Results marked with † are statistically significantly ($p<0.05$) better than the ones of the best-performing baseline (‡ represents significance with $p<0.01$). * indicates statistically significantly better than **QSD** ($p<0.05$).

Methods	Precision	Recall	F_1-score
Overlap	0.63	0.48	0.55
VSM	0.23	0.17	0.20
QSD-NT	0.46	0.34	0.39
QSD	0.66†	0.50†	0.57†
SSD	**0.72‡***	**0.54‡***	**0.61‡***

A. Importance of Aligned Differences

We can observe from Tab. I that **Overlap** is quite competitive. Yet, it still performs worse than the proposed methods indicating that the commonality is not enough for effective similarity comparison. According to psychological studies [5], [35], aligned differences are actually central to the comparison process of humans and have been found very influential in decision making. Especially they are important in heterogeneous spaces which tend to have few commonalities. For example, for the query: *Arnold Schwarzenegger* vs. *Arnold Schwarzenegger*, not only obvious and expected commonalities (e.g., `Hollywood` :: `Hollywood`) should be output but also the fact that *Schwarzenegger* shifted his career focus from being a movie star to serving as the governor of California (i.e., from film-making to politics). Note that both our proposed methods selected the pair `Bustamante` :: `Stallone` (see Tab. II) which points to this fact (`Bustamante` as a key political competitor of *Schwarzenegger* in [2002, 2007] and `Stallone` as a major rival in the film industry in [1987, 1991])[4]. However, **Overlap** cannot detect this change as it can only map a movie-oriented term from one time to the same movie-oriented term at the other time (due to its time-invariance).

B. Limitation of Traditional Vector Space Models

The next observation is that the task is quite difficult as evidenced by the poor performance of the adjusted Vector Space Model approach (**VSM**). **VSM** measures term similarity based on the co-occurrence assumption, rather than based on term semantics as in the case of our proposed methods, which utilize word embeddings. It works well for some easy lexical patterns, such as hypernyms or is-a relationships, e.g., "Apple is the company of iPod", "Blatter is the president of FIFA", "Havelange is the president of the world governing body of soccer" (see column **VSM** in Tab. II for ID#s: 1-1, 3-3, 3-4). However many times, more difficult relations are expressed in text which cannot be represented by simple lexical patterns, such as ID#s: 2-2, 3-2, 4-2 in Tab. II[5].

C. Necessity of Transformation when Heterogeneous Spaces

In realistic scenarios, entity comparison is often conducted across heterogeneous spaces such as across different time periods as in our experiments. **QSD-NT** however essentially assumes a static world in which each term is supposed to retain its semantics across different domains (or has the same "position" in a single common vector space created using the combined documents from both the domains, i.e., different time periods in our case). Yet, many terms tend to change their meaning and usage in different domains. Thus, their relative "positions" wrt. to other terms should change, too. Without the transformation, the information on the relative changes of term positions in a vector space is lost. This can explain why **QSD-NT** does not select `Apple`::`Sony` pair (Tab. II). Since the two companies existed in both the time periods, **QSD-NT** simply selects the incorrect pair: `Sony`::`Sony`, instead.

D. Commonalities vs. Aligned Differences

Tab. III shows the detailed performance of tested methods on detecting *commonalities* and *aligned differences*. Not surprisingly, **Overlap** outperforms other methods in commonality detection. As for *aligned differences*, the proposed methods are statistically significantly different ($p<0.01$) from the best baseline **QSD-NT** both in terms of precision and recall (see Al.diff. columns in Tab. III).

E. Usefulness of Systematicity

According to Tab. I, **SSD** statistically significantly outperforms **QSD** across all the measures. This emphasizes the importance of applying the concept of systematicity for aggregating structural alignments among all the candidate term pairs. **SSD** demonstrates capability of detecting term pairs which have good alignment with the entity context by considering the relation among all the candidate term pairs. This can be seen for the query *Sepp Blatter* vs. *Joao Havenlange* (FIFA presidents during [2002, 2007] and [1987, 1991], respectively) shown in Tab. II. **SSD** selects `Zidane` :: `Vautrot` since both the players caused controversial issues in the 2006 and 1990 World Cups, respectively, significantly impacting FIFA's reputation. We can also see in Tab. III that **SSD** outperforms **QSD** in both commonalities and aligned differences.

F. Query Types

In Tab. IV, we compare the performance of the proposed methods over the three different query types. We can observe that F_1 remains relatively stable for different query types, although the precision of locations is lower than the one for the other two query types. This might be due to the higher topic diversity of locations when compared to more specific objects and persons.

G. Examples of Supporting Sentences

Finally, in Tab. V we also show the examples of supporting sentences created for the term pairs displayed in Tab. II. Note that in order to enlarge the candidate pool of supporting sentences, we detect different names of the same entity (see, for example, ID# 2-1,3-2,3-3 in Tab. V). Specifically, we check (1) if a term is a substring (e.g., *Schwarzenegger*) of a query (e.g., *Arnold Schwarzenegger*), and (2) if the term and the query are close in semantic vector space (i.e., the term should be among the top 5 nearest neighbors of the query).

[4] It is also explained by supplementary sentences in Tab. V-(2-1).

[5] The corresponding supporting sentences in Tab. V explain such relations.

TABLE II: Example results. For each query we list four ground truth examples (together with manually added labels shown in parentheses indicating how the examples relate to their entities) selected by annotators: two examples of *aligned differences* and two examples of *commonalities*. ✓indicates that a given term pair was detected by a particular method.

ID#	Queried entities & correct term pairs	Overlap	VSM	QSD-NT	QSD	SSD
	iPod[2002,2007] vs. ***Walkman***[1987,1991]					
1-1	Apple :: Sony (company)		✓		✓	✓
1-2	MP3 :: cassette (media)				✓	✓
1-3	portable :: portable (characteristic)	✓			✓	✓
1-4	music :: music (usage)	✓				✓
	Arnold Schwarzenegger[2002,2007] vs. ***Arnold Schwarzenegger***[1987,1991]					
2-1	Bustamante :: Stallone (competitor)				✓	✓
2-2	Californians :: moviegoers (supporter)			✓	✓	✓
2-3	Hollywood :: Hollywood (industry)	✓			✓	✓
2-4	Terminator :: Terminator (movie)	✓		✓	✓	✓
	Sepp Blatter[2002,2007] vs. ***Joao Havenlange***[1987,1991]					
3-1	Klinsmann :: Osim (coach)				✓	✓
3-2	Zidane :: Vautrot (controversy)					✓
3-3	FIFA :: FIFA (organization)	✓	✓	✓	✓	✓
3-4	soccer :: soccer (field)	✓	✓	✓	✓	✓
	Germany[2002,2007] vs. ***East Germany***[1987,1991]					
4-1	Gerhard Schröder :: Hans Modrow (heads of country)				✓	✓
4-2	European :: Soviet (union)			✓		
4-3	Berlin :: Berlin (capital)	✓		✓	✓	✓
4-4	Germans :: Germans (citizen)	✓		✓	✓	✓

TABLE III: Results of detecting *commonalities* (Com.) and *aligned differences* (Al.diff.). * indicates results statistically significantly (p<0.01) better than the best performing baseline. † indicates those better than **QSD** method (p<0.05).

Method	**Precision**		**Recall**		**F$_1$-score**	
	Com.	**Al.diff.**	**Com.**	**Al.diff.**	**Com.**	**Al.diff.**
Overlap	**0.63**†	0.00	**0.99**†	0.00	**0.77**†	0.00
VSM	0.13	0.11	0.18	0.15	0.15	0.13
QSD-NT	0.25	0.22	0.37	0.30	0.30	0.25
QSD	0.29	0.37*	0.43	0.54*	0.35	0.44*
SSD	0.32†	**0.39***	0.49†	**0.57***†	0.39†	**0.47***†

TABLE IV: Performance over different types of queries (Precision/Recall/F$_1$-score).

Method	***Objects***	***Persons***	***Locations***
QSD	.70/.47/.56	.65/.51/.56	.62/.53/.56
SSD	.75/.50/.60	.72/.56/.60	.65/.56/.62

IX. Conclusions and Future Work

In this paper, motivated by the lack of evidences to support the similarity computation in word embedding, we have introduced a novel problem of explaining the similarity of terms by finding their commonalities and aligned differences. In particular, we have proposed two unsupervised methods to solve this task and we have successfully demonstrated their effectiveness in the case of heterogeneous spaces. In the future, we wish to test our methods in other scenarios, in particular, when using different corpora and with different word embedding techniques as well as we will experiment with incorporating external ontologies or aspect mining techniques into our methods.

References

[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. of ICLR Workshop*, 2013.

[2] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation." in *EMNLP*, vol. 14, 2014, pp. 1532–43.

[3] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representation of phrases and their compositionality," in *Proc. of NIPS*, 2013, pp. 3111–3119.

[4] N. Tahmasebi, G. Gossen, N. Kanhabua, H. Holzmann, and T. Risse, "Neer: An unsupervised method for named entity evolution recognition," in *Proc. of Coling*, 2012, pp. 2553–2568.

[5] D. Gentner and A. B. Markman, "Structure mapping in analogy and similarity," *American psychologist*, vol. 52, no. 1, p. 45, 1997.

[6] O. Kolomiyets and M.-F. Moens, "A survey on question answering technology from an information retrieval perspective," *Inf. Sci.*, vol. 181, no. 24, pp. 5412–5434, Dec. 2011.

[7] V. Singh and S. K. Dwivedi, "Question answering: A survey of research, techniques and issues," *International Journal of Information Retrieval Research*, vol. 4, no. 3, pp. 14–33, 2014.

[8] C. J. Halperin, G. Y. R. J. Loewenberg, and et al., "Comparative history in theory and practice: A discussion," *The American Historical Review*, vol. 87, no. 1, pp. 123–143, 1982.

[9] M. L. Wilson, M. C. schraefel, and R. W. White, "Evaluating advanced search interfaces using established information-seeking models," *J. Am. Soc. Inf. Sci. Technol.*, vol. 60, no. 7, pp. 1407–1422, Jul. 2009.

[10] B. Kim, J. Scott, and S. Kim, "Exploring digital libraries through visual interfaces," *In: Kuo Hung Huang: "Digital Libraries - Methods and Applications"*, pp. 123–136, 2011.

[11] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A core of semantic knowledge," in *Proc. of WWW*, 2007, pp. 697–706.

[12] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann, "Dbpedia - a crystallization point for the web of data," *Web Semant.*, vol. 7, no. 3, pp. 154–165, Sep. 2009.

[13] E. Sandhaus, "The new york times annotated corpus overview," *The New York Times Company, Research & Develop.*, pp. 1–22, 2008.

[14] B. Falkenhainer, K. D. Forbus, and D. Gentner, "The structure-mapping engine: Algorithm and examples," *Artificial intelligence*, vol. 41, no. 1, pp. 1–63, 1989.

[15] P. D. Turney, "The latent relation mapping engine: Algorithm and experiments," *Journal of Artificial Intelligence Research*, pp. 615–655, 2008.

[16] P. Turney, "Expressing implicit semantic relations without supervision," *CoRR*, 2006.

[17] D. Gentner, "Structure-mapping: A theoretical framework for analogy*," *Cognitive science*, vol. 7, no. 2, pp. 155–170, 1983.

[18] D. Bollegala, T. Goto, N. T. Duc, and M. Ishizuka, "Improving relational similarity measurement using symmetries in proportional word analogies," *Information Processing & Management*, vol. 49, no. 1, pp. 355–369, 2013.

[19] D. Bollegala, M. Kusumoto, Y. Yoshida, and K.-I. Kawarabayashi, "Mining for analogous tuples from an entity-relation graph," in *Proc. of AAAI*, 2013, pp. 2064–2077.

[20] M. P. Kato, H. Ohshima, S. Oyama, and K. Tanaka, "Query by analogical example: relational search using web search engine indices," in *Proc. of CIKM*, 2009, pp. 27–36.

[21] T. Veale, "Wordnet sits the sat-a knowledge-based approach to lexical analogy," in *ECAI*, vol. 16, 2004, pp. 606–612.

TABLE V: Sentences generated for example term pairs listed in Tab.II using the method described in Sec.VI-B. Bold words are entities, while bold and underlined words represent selected terms.

ID#	**Supporting Sentences:**
1-1	- Apple issued its angry statement ... play them on **Apple**'s popular **iPod** portable music players, as well as players using windows media player... - **Sony** is staking much of its future in "personal video" a new genre of products, based on small, eight-millimeter videocassetes, of which the **Walkman** is among the first.
1-2	- **IPod**s, the most popular music players with more than 70 percent of the american market, can play **MP3** music files, a popular digital audio compression format. - the Sony corporation is staking much of its future on a new genre of products like the video **Walkman**, a tiny television-video **cassette** recorder, that are based on the small, eight-millimeter videotape that Sony manufactures.
1-3	- Apple introduces iTunes software for podcasts apple computer made available a version of its itunes software intended for subscribing to podcasts, audio recordings that can be downloaded to an **iPod** for **portable** listening. - Finally, l988 marks the year of the video **Walkman**, Sony's incredibly compact combination of an 8-millimeter VCR with a three-inch flat-screen color television set in a **portable** package no bigger than an ordinary VHS cassette.
1-4	- Apple issued its angry statement... play them on Apple's popular **iPod** portable **music** players, as well as players using windows media player... - the Sony **Walkman** helped usher in a new category known as "personal audio" products, which allow people to take high quality **music** with them anywhere.
2-1	- **Bustamante**, a democrat, is the leading candidate to replace him if the recall succeeds, holding a narrow margin over his closest competitor, **Arnold Schwarzenegger**, a republican. - In theatrical-release films, the big roles, and the gigantic salaries, are dominated by fellows with names like Newman, Redford, **Stallone**, **Schwarzenegger** and Costner.
2-2	- That became clear on Saturday when one of the state's top Democrats, Attorney General Bill Lockyer, told a conference here that he not only understood why so many **Californians** had voted for **Arnold Schwarzenegger**, the Republican governor-elect, but also that he had voted for Mr. Schwarzenegger. - (None)
2-3	- "R-rated: republicans in **Hollywood**", a documentary tonight on AMC, examines politics in the entertainment industry at a time when the white house and congress are in republican hands, conservatives dominate the supreme court, **Arnold Schwarzenegger** is governor of California and mel gibson's "passion of the christ" triumphed at the box office. - Furst also designed planet **Hollywood**, the $15 million Manhattan restaurant owned by the actors **Arnold Schwarzenegger**, Sylvester Stallone and Bruce Willis.
2-4	- Situated in the heart of downtown, it is called arnold classic, after **Arnold Schwarzenegger**, the champion bodybuilder turned **Terminator** turned governor of california. - Already, one big-budget offering, tri-star's $50 million "hudson hawk,"... splashy action-adventure films as the $45 million "robin hood: prince of thieves," starring kevin costner, and the $85 million "**Terminator** 2: judgment day", with **Arnold Schwarzenegger**.
3-1	- (None) - (None)
3-2	- **Blatter**, the president of FIFA, soccer's governing body, said that the golden ball award **zidane** received as the world cup's most outstanding player may be revoked; the award was voted on by members of the news media. - Working in front of 73,780 fans in milan, plus assorted heads of state, to say nothing of **Joao Havelange** himself, **Vautrot** came out flashing yellow cards like somebody hawking handbills for a new fast-food stand.
3-3	- **Blatter**, the president of soccer's world governing body, **FIFA**, reiterated yesterday that South Africa would be ready to stage the 2010 World Cup. - **Joao Havelange**, the president of **FIFA**, the world governing body for soccer, reaffirmed at a news conference that there was "no debate" about the 1994 world cup being played in the united states.
3-4	- The collapse of the marketing partner of FIFA, **soccer**'s world governing body, has had an enormous impact on FIFA and its competitions, not to mention the position of its president, **Sepp Blatter** of Switzerland. - After Argentina proved itself better than Italy at penalty kicks in the semifinals of the world cup last Tuesday night, the president of the world governing body of **soccer**, **Joao Havelange**, received a fax from a distraught italian woman complaining that her country should not have been eliminated in such an unfair manner.
4-1	- 10 about the appointment of **Gerhard Schröder**, the former chancellor of **Germany**, as chairman of the northern European gas pipeline company, misstated the name of a gas field being developed in Russia, and the size of its reserves. - Those taking part in the debate were president Wojciech Jaruzelski of Poland, prime minister **Hans Modrow** of **East Germany**, prime minister Marian Calfa of Czechoslovakia, prime minister andrei lukanov of Bulgaria and deputy prime minister Peter Medgyessy of Hungary.
4-2	- When Tony Blair, the British prime minister, meets president Jacques Chirac of France and chancellor Gerhard Schröder of **Germany** at a **European** union summit on June 16, they will begin what amounts to the great European salvage operation: an attempt to define some new idea for a Europe ... - (None)
4-3	- **Germany**'s economics and labor minister, Wolfgang Clement, said that **Berlin** would defend the company's jobs in Germany. - In 1970, the United states, Britain, France and the Soviet Union... reached an agreement aimed at easing tensions, allowing the west easier access to west **Berlin**...
4-4	- He is the leader in his locality in northwestern **Germany** of the association of expellees, an organization claiming a million members that represents the interests of the estimated 12 million to 13 million **Germans** who were expelled from Poland and other countries when world war ii ended. - But the unrelenting surge of more than 2,000 east german emigrants into west germany daily has added urgency to the plans for creating a currency union that would integrate **East Germany** into the west German economy, providing the east **Germans** with the advantages of the strong West German capitalist economy...

[22] R. Ratcliff and G. McKoon, "Similarity information versus relational information: Differences in the time course of retrieval," *Cognitive Psychology*, vol. 21, no. 2, pp. 139–155, 1989.

[23] R. Mihalcea and P. Tarau, "Textrank: Bringing order into texts," in *Proc. of EMNLP*, 2004, pp. 404–411.

[24] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *Proc. of SIGIR*, 1998, pp. 335–336.

[25] K. Berberich, S. J. Bedathur, M. Sozio, and G. Weikum, "Bridging the terminology gap in web archive search," in *Proc. of WebDB*, 2009.

[26] Y. Zhang, A. Jatowt, S. S. Bhowmick, and K. Tanaka, "Omnia mutantur, nihil interit: Connecting past with present by finding corresponding terms across time," in *Proc. of ACL*, 2015, pp. 645–655.

[27] V. Kulkarni, R. Al-Rfou, B. Perozzi, and S. Skiena, "Statistically significant detection of linguistic change," in *Proc. of WWW*, 2015, pp. 625–635.

[28] A. Jatowt and K. Duh, "A framework for analyzing semantic change of words across time," in *Proc. of JCDL*, 2014, pp. 229–238.

[29] M. Pargel, Q. D. Atkinson, and A. Meade, "Frequency of word-use predicts rates of lexical evolution throughout indo-european history," *Nature*, vol. 449, pp. 717–720, 2007.

[30] E. Lieberman, J. B. Michel, J. Jackson, T. Tang, and M. A. Nowak, "Quantifying the evolutionary dynamics of language," *Nature*, pp. 713–716, 2007.

[31] J. K. Sparck and V. C. Rijsbergen, *Report on the need for and provision of an ideal information retrieval test collection*. Computer Laboratory, University of Cambridge: British Library Research and Development Report 5266, 1975.

[32] J. L. Fleiss, "Measuring nominal scale agreement among many raters," *Psychological Bulletin*, vol. 76, no. 5, pp. 378–382, 1971.

[33] R. J. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, pp. 159–174, 1977.

[34] P. D. Turney and M. L. Littman, "Corpus-based learning of analogies and semantic relations," *Machine Learning*, vol. 60, no. 1-3, pp. 251–278, 2005.

[35] P. G. Lindemann and A. B. Markman, "Alignability and attribute importance in choice," in *Proc. of 8th annual meeting of the Cognitive Science Society*, 1996, pp. 358–363.

TABLE VI: Top 20 terms output by each method for selected query pairs. Pairs in bold font are judged as correct by annotators.

Overlap	**VSM**	**QSD-NT**	**QSD**	**SSD**
iPod[2002,2007], *Walkman*[1987,1991]				
Sony:: Sony	**Apple**:: **Sony**	iTunes:: Sony	**Apple**:: **Sony**	**Apple**:: **Sony**
audio:: **audio**	iTunes:: agers	Sony:: Sony	**MP3**:: **cassette**	**MP3**:: **cassette**
digital:: digital	MP3:: ohga	Apple:: Nintendo	iTunes:: cassettes	iTunes:: cassette
portable:: **portable**	nano:: schrodinger	MP3:: discman	**nano**:: **discman**	**pogue**:: **norio**
cd:: cd	gigabyte:: discman	zune:: cassette	**pogue**:: **norio**	**zune**:: **discman**
video:: video	Sony:: norio	portable:: cassettes	**gigabyte**:: **millimeter**	**gigabyte**:: **millimeter**
headphones:: **headphones**	portable:: schulhof	**audio**:: **audio**	zune:: rechargeables	nano:: rechargeables
music:: **music**	Zune:: Yetnikoff	digital:: dat	**Sony**:: **Nintendo**	**Sony**:: **Nintendo**
stereo:: stereo	**audio**:: **audio**	nano:: disks	digital:: digital	**portable**:: **portable**
electronics:: **electronics**	music:: morita	podcasting:: video	**portable**:: **portable**	digital:: digital
Los Angeles:: Los Angeles	digital:: digital	devices:: rechargeables	audio:: video	cd:: cd
computer:: computer	Macintosh:: tasman	download:: earphones	Macintosh:: rom	video:: video
disk:: disk	wifi:: teen	music:: cd	cd:: cassette	**headphones**:: **headphones**
inch:: inch	cd:: cassette	gigabyte:: disk	version:: agers	version:: agers
screen:: screen	mac:: zazen	podcasts:: vcr	music:: prerecorded	audio:: cassettes
recorder:: **recorder**	**video**:: **cassettes**	**headphones**:: **headphones**	**headphones**:: **headphones**	**music**:: **music**
compact:: **compact**	pc:: Nintendo	**video**:: **tape**	zen:: schrodinger	mac:: rom
software:: software	computer:: rom	bluetooth:: stereo	bluetooth:: vhs	zen:: schrodinger
recording:: **recording**	download:: video	pc:: portable	**video**:: **tape**	**Microsoft**:: **Matsushita**
format:: format	**headphones**:: **headphones**	imac:: compact	**Microsoft**:: **Matsushita**	**electronics**:: **electronics**
Vladimir Putin[2002,2007], *Boris Yeltsin*[1987,1991]				
anti:: anti	Russia:: Mikhail	Russia:: republics	Russian:: Mikhail	**Boris Yeltsin**:: **Mikhail Gorbachev**
soviet:: soviet	Boris Yeltsin:: anti	Lugovoi:: Gorbachev	**Boris Yeltsin**:: **Mikhail Gorbachev**	Russian:: Mikhail Gorbachev
Russia:: **Russia**	Soviet:: Mikhail Gorbachev	Russian:: Soviet	**Russia**:: **Soviet**	**Russia**:: **Soviet**
Mikhail Gorbachev:: Mikhail Gorbachev	Russian:: Soviet	Boris Yeltsin:: Engver	**nuclear**:: **anti**	Lugovoi:: engver
Mikhail Gorbachev:: Mikhail Gorbachev	Kremlin:: engver	Chechnya:: Moscow	Lugovoi:: engver	**nuclear**:: **anti**
Russian:: **Russian**	petersburg:: coup	Mikhail Khodorkovsky:: Mikhail Gorbachev	Soviet:: Russian	Kremlin:: coup
Moscow:: **Moscow**	**Moscow**:: **Moscow**	**Vagit Alekperov**:: **Vadim Bakatin**	Kremlin:: coup	Soviet:: Russian
Kremlin:: **Kremlin**	Chechnya:: Russian	Kremlin:: Russian	**Moscow**:: **Moscow**	**Moscow**:: **Moscow**
United States:: **United States**	Mikhail Khodorkovsky:: union	**oligarchs**:: **reformers**	united:: union	united:: union
Bush:: **Bush**	Mikhail Gorbachev:: republics	Kovtun:: Kravchuk	Chechnya:: republics	**oligarchs**:: **republics**
Ukraine:: **Ukraine**	**Chechen**:: **Russia**	Chechen:: communist	Chechen:: communist	**Bush**:: **Bush**
communist:: **communist**	lugovoi:: liners	**Russians**:: **Baltics**	**oligarchs**:: **reformers**	**Chechnya**:: **Russia**
nuclear:: **nuclear**	anti:: editorials	**Ukraine**:: **Ukraine**	**Ukraine**:: **Russia**	Chechen:: communist
union:: union	alekperov:: a1	Moscow:: Russia	Litvinenko:: Bakatin	**Vagit Alekperov**:: **Vadim Bakatin**
Stalin:: **Stalin**	**oligarchs**:: **communist**	yukos:: kremlin	**Bush**:: **Bush**	lehrer:: schuster
president:: **president**	russians:: republic	chechens:: plotters	russians:: plotters	russians:: plotters
republics:: republics	**United States**:: **United States**	Stalin:: Perestroika	lehrer:: schuster	litvinenko:: kravchuk
russians:: **russians**	Bush:: reform	soviet:: union	kovtun:: kravchuk	**president**:: **president**
minister:: minister	maskhadov:: plotters	yushchenko:: ryzhkov	**president**:: **president**	**Stalin**:: **Stalin**
republic:: republic	**Ukraine**:: **Ukraine**	**Bush**:: **Bush**	**Stalin**:: **Stalin**	**Ukraine**:: **Ukraine**
Google[2002,2007], *IBM*[1987,1991]				
Microsoft:: **Microsoft**	Yahoo:: OS	Yahoo:: computer	silicon:: megabit	**Yahoo**:: **Compaq**
silicon:: silicon	search:: megabit	Microsoft:: mainframe	**web**:: **computer**	**Microsoft**:: **Microsoft**
software:: **software**	**web**:: **mainframe**	**users**:: **computers**	**search**:: **pc**	silicon:: megabit
computer:: computer	brin:: computer	**msn**:: **mainframes**	**users**:: **machines**	**web**:: **computer**
users:: users	**Microsoft**:: **Microsoft**	internet:: software	Yahoo:: mainframe	**Brin**:: **Kuehler**
desktop:: desktop	silicon:: machines	**Gmail**:: kuehlerpc	**Gmail**:: **computers**	**users**:: **machines**
computers:: computers	**msn**:: **pc**	**AOL**:: **Compaq**	**Brin**:: **Kuehler**	**search**:: **pc**
company:: **company**	**users**:: **computers**	**Youtube**:: **Microsoft**	msn:: software	**msn**:: **mainframe**
pc:: pc	internet:: compaq	browser:: os	internet:: desktop	**Gmail**:: **computers**
digital:: digital	Gmail:: software	software:: machines	Youtube:: digital	**Semel**:: **Akers**
user:: user	youtube:: ps	**search**:: **users**	**Microsoft**:: **Intel**	**software**:: **software**
technology:: **technology**	sergey:: desktop	online:: desktop	**Semel**:: **Akers**	wifi:: perkin
Apple:: **Apple**	AOL:: perkin	**Baidu**:: **Apple**	**Baidu**:: **ROLM**	**Baidu**:: **ROLM**
computing:: computing	ads:: Kuehler	desktop:: ps	**browser**:: **DOS**	**AOL**:: **Intel**
vice:: vice	Baidu:: mainframes	Semel:: Cannavino	company:: microsystems	**browser**:: **desktop**
companies:: companies	engine:: microsystems	technology:: risc	wifi:: perkin	company:: microsystems
business:: business	**Semel**:: **Akers**	doubleclick:: Intel	**engine**:: **chip**	**Schmidt**:: **Geller**
Intel:: Intel	online:: Apple	click:: Macintosh	**engines**:: **mainframes**	Youtube:: digital
San Francisco:: San Francisco	wifi:: intel	**web**:: **user**	copyrighted:: chips	click:: DOS
market:: market	sites:: dos	**user**:: **microprocessor**	user:: macintosh	**engine**:: **chip**
Arnold Schwarzenegger[2002,2007], *Arnold Schwarzenegger*[1987,1991]				
Terminator:: **Terminator**	huffington:: Terminator	**Terminator**:: **Terminator**	**Los Angeles**:: **Los Angeles**	**Terminator**:: **Terminator**
Los Angeles:: **Los Angeles**	Bustamante:: Carolco	**Shriver**:: **Shriver**	**Bustamante**:: **Stallone**	**Los Angeles**:: **Los Angeles**
Shriver:: **Shriver**	**Los Angeles**:: **Los Angeles**	**Mcclintock**:: **Murphy**	**Mcclintock**:: **Seagal**	**Bustamante**:: **Stallone**
Total Recall:: **Total Recall**	California:: tri	movie:: Carolco	**Huffington**:: **Grazer**	**Huffington**:: **Seagal**
movie:: **movie**	**Mcclintock**:: **Stallone**	**Total Recall**:: **Total Recall**	**Terminator**:: **Terminator**	**Mcclintock**:: **Grazer**
Hollywood:: **Hollywood**	Terminator:: film	bodybuilder:: cop	shriver:: dustin	**San Francisco**:: **Los Angeles**
film:: **film**	Shriver:: movie	Hollywood:: film	**San Francisco**:: **Los Angeles**	Shriver:: Dustin
movies:: **movies**	Total Recall:: Seagal	**californians**:: **moviegoers**	angelides:: costner	**Angelides**:: **Costner**
Bush:: Bush	governor:: movies	**Davis**:: **Willis**	**Davis**:: **Devito**	**Total Recall**:: **Total Recall**
United States:: **United States**	gov:: wildman	actor:: movie	**Leno**:: **Bogosian**	**Davis**:: **Devito**
star:: **star**	angelides:: studios	**movies**:: **movies**	**Total Recall**:: **Total Recall**	**Leno**:: **Bogosian**
actor:: **actor**	**San Francisco**:: **Los Angeles**	celebrity:: Hollywood	bodybuilder:: cop	bodybuilder:: cop
fitness:: fitness	Davis:: films	California:: Los Angeles	California:: Columbia	**Hollywood**:: **Hollywood**
United States:: **United States**	californians:: rambo	film:: films	**Bush**:: **Katzenberg**	**movie**:: **movie**
television:: television	anti:: teen	governor:: budget	**Hollywood**:: **Hollywood**	**californians**:: **moviegoers**
vice:: vice	Ueberroth:: Grazer	hero:: rambo	**californians**:: **moviegoers**	**California**:: **Columbia**
Reagan:: Reagan	lockyer:: Hollywood	television:: studios	**movie**:: **film**	**film**:: **film**
president:: president	sacramento:: bogosian	Sacramento:: Los Angeles	governor:: Bush	governor:: Bush
office:: office	bodybuilder:: dustin	**star**:: **star**	gov:: John	**movies**:: **movies**
million:: million	ballot:: robocop	voters:: people	francisco:: walt	company:: Carolco