Data Science and Engineering (2019) 4:208–222
https://doi.org/10.1007/s41019-019-00102-3

Mapping Entity Sets in News Archives Across Time

Yijun Duan[1] · **Adam Jatowt**[1] · **Sourav S. Bhowmick**[2] · **Masatoshi Yoshikawa**[1]

Received: 25 June 2019 / Revised: 20 August 2019 / Accepted: 23 August 2019 / Published online: 9 September 2019

Abstract
We propose a novel way of utilizing and accessing information stored in news archives as well as a new style of investigating the history. Our idea is to automatically generate similar entity pairs given two sets of entities, one from the past and one representing the present. This allows performing entity-oriented mapping between different times. We introduce an effective method to solve the aforementioned task based on a concise integer linear programming framework. In particular, our model first conducts typicality analysis to estimate entity representativeness. It next constructs orthogonal transformation between the two entity collections. The result is a set of typical across-time comparables. We demonstrate the effectiveness of our approach on the New York Times dataset through both qualitative and quantitative tests.

Keywords Comparable entity mining · Typicality analysis · Temporal embeddings alignment · Integer linear programming

1 Introduction

Named entities are first class citizens in news and typically attract a lot of focus in any news article analysis. Therefore, automatic approaches toward effective named entity detection, disambiguation, linking and understanding have gathered much attention and have been studied since long ago. This is also true for the case of archival news article collections, albeit ancient documents typically add additional challenges for those tasks. The way in which history is studied and taught is also often entity centric and event centric.

One of common objectives of studying the history is to compare it with the present for drawing informative, novel or interesting conclusions. Accordingly, in such across-time scenarios an entity-to-entity comparison is an intuitive approach alongside event-to-event comparison. Imagine a journalist who aims to undertake a study of a certain period in the past in order to gain insights concerning similarity between the present political scene and the one in that period. In this case, an entity-centric comparison could provide him or her with a novel outlook on the present political elites and lead to drawing new conclusions and discussions with regard to wider political scene. Opinions on the continuity, change or analogy could be then formed or supported. Across-time entity-centered contrast may be also useful for scholars (e.g., historians or social scientists) studying different views of the past. For example, automatic comparison of household appliances in 1970s–1980s with those used in 2000s–2010s could facilitate work for scientists interested in the history and evolution of technology used at home.

In general, the determination of corresponding entity pairs is beneficial for in-depth understanding of the relation between the past and present and leads to the improved comprehension of our history and heritage. Furthermore, it can also support generation of forecasts according to the intuition that future events may resemble past ones based on the correspondence and similarity of their actors, i.e., entities. This kind of across-time comparison could be also seen as a special type of data integration problem, as it essentially consists of combining data that reside in different sources (entities in distant periods) and providing users with a unified view of them (generated mappings). We note, however, that comparing entire entity sets manually is not easy since

✉ Yijun Duan
minsak1020@gmail.com

Adam Jatowt
adam@dl.kuis.kyoto-u.ac.jp

Sourav S. Bhowmick
assourav@ntu.edu.sg

Masatoshi Yoshikawa
yoshikawa@i.kyoto-u.ac.jp

[1] Graduate School of Informatics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan

[2] School of Computer Science and Engineering, Nanyang Technological University, Singapore, Singapore

people tend to possess limited knowledge about things from the past and lack historical perspective, necessary for comparisons involving longer time gaps. Another reason is possibly large size of entity sets, which can be also diverse and complex, demanding much cognitive effort. The objective of this work is then to propose automatic approaches for comparison of entity sets across time.

A natural method of set comparison is to find pairs of corresponding and representative entities of the both sets. Actually, example-based learning is an effective strategy that humans tend to adopt. Good examples help to grasp difficult concepts or complex entity categories in more effective way than high-level feature descriptions. Therefore, in our scenario, given two input collections of entities from different times (e.g., present politicians and ones from 1970s–1980s), we would aim to automatically generate a diversified set of corresponding entity pairs (e.g., US Presidents: *Donald Trump* and *Ronald Reagan*, Russian Presidents: *Vladimir Putin* and *Mikhail Gorbachev* or others less obvious examples in the case of politicians comparison). Given the generated entity mappings, a journalist, scientist or other professional could then be better equipped to further match events or entire periods in which those entities played key roles (e.g., matching political scenes of different times) in order to formulate or support various theories and analogies.

The problem of automatically generating comparable entity pairs is difficult due to the following challenges: (1) it is non-trivial to design approaches for representing across-time entity correspondence, especially, in cases when the entity sets come from time periods separated by long time gaps. The development of such correspondences needs to take into consideration differences in general contexts of the different, studied time periods. Collecting training data for establishing such connections is also challenging. (2) Effective comparison should involve typical entity representatives as these are usually associated with better representativeness of features and are less likely to cause misunderstanding. To give an example from biology, for an effective comparison of mammals with another animal class, one would prefer typical mammals like lions over atypical ones like platypuses (which lay eggs instead of giving birth). (3) Lastly, the input entity sets can be quite large and diverse (e.g., celebrities). Hence, the typical-example selection strategy would need to take into account any detected latent subgroups. Based on this, rather than proposing a single entity pair as an output, a set of pairs (each for every latent subgroup) would be preferred. An effective method then needs to output an optimal subset of all possible pairs, which would be composed of both typical and temporally comparable entities.

To approach the above-mentioned challenges, we introduce a novel technique for generating *typical across-time comparables*. First of all, we base the measurement of *entity typicality* on the findings from psychology and cognitive sciences [6, 14, 39]. In particular, an entity is considered typical in a diverse set if it is representative within a significant subset of that set. We next formulate the measurement of across-time entity comparability by first aligning vector spaces derived from the compared periods and then finding corresponding terms. For this, we adopt the distributed vector representation [27] to represent the context vectors of entities and we learn linear and orthogonal transformations between two vector spaces of input collections for establishing across-time entity correspondence. Lastly, inspired by the popular affinity propagation (AP) algorithm [11], we propose a concise joint integer linear programming framework (J-ILP) which detects typical entities (subsequently called exemplars) and outputs comparable pairs from the detected exemplars.

We perform experiments on the New York Times (NYT) Annotated Corpus [34], which has been frequently used to evaluate different researches that focus on temporal information processing or extraction in document archives [3]. The experiments demonstrate that the proposed method outperforms the baselines by 58.7% percent on average.

This paper is an extended version of our previous work [5]. We improve the previous work by (1) conducting extensive experiments on additional time periods of NYT dataset with new detailed analysis; (2) introducing additional novel metrics (*typicality*, *comparability* and *product*) for providing more evidence to demonstrate the effectiveness of proposed methods. Finally, (3) we discuss the key factors which can affect the model performance in detail.

2 Problem Definition

Formally, given two sets of entities denoted by D_A and D_B, where D_A and D_B come from different time periods T_A and T_B, respectively ($T_A \cap T_B = \emptyset$ and, typically T_A represents some period in the past while T_B represents more present time period), the task is to discover m comparable entity pairs $P = [p_1, p_2, \cdots, p_m]$ to form a concise subset conveying the most important comparisons, where $p_i = (e_i^A, e_i^B)$. e_i^A and e_i^B are entities from D_A and D_B, respectively (see Fig. 1 for an illustration of our research problem). The pairs should have good quality, i.e., each entity should be representative in its set, and entities within the same pair should be temporally comparable. Moreover, the set of selected entities should cover as many subgroups as possible to avoid redundancy.

3 Estimation of Entity Typicality

Learning from examples is an effective strategy extensively adopted in cognition and education [14]. Good examples should be, however, typical. In this work, we apply the

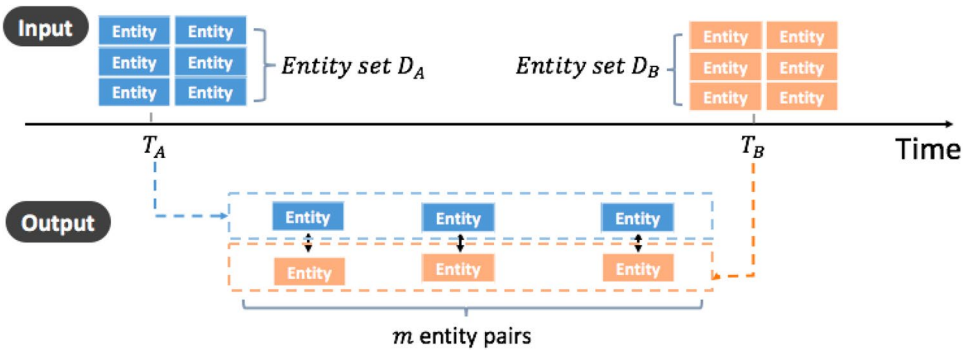

Fig. 1 Illustration of our research problem

strategy of using typical examples for discovering comparable entity pairs. We denote the typicality of an entity e with regard to a set of entities S as $\text{Typ}(e, S)$. The entities to be selected for comparison should be typical in their sets; namely, $\text{Typ}(e_i^A, D_A)$ and $\text{Typ}(e_i^B, D_B)$ should be as high as possible when $p_i = (e_i^A, e_i^B)$ is a selected entity pair.

As suggested by the previous research in typicality analysis [14], an entity e in a set of entities S is typical, if it is likely to appear in S. We denote the likelihood of an entity e given a set of entities S by $L(e|S)$ (to be defined soon). However, it is not appropriate to simply use $L(e|S)$ as an estimator of typicality $\text{Typ}(e, S)$ considering the characteristics of our task. First of all, the collections of entities for comparison can be very complex, and thus, they may cover many different kinds of entities. For example, if we want to compare US scientists across time, each of entity collections will include multiple kinds of entities such as mathematicians, physicists, chemists and so on. It is then very difficult for a single entity to represent all of them. In addition, different entity kinds vary in their significance. For instance, "physicists" are far more common than "entomologists". Naturally, entities typical in a salient entity subset should be more important than those belonging to small subsets.

Given a set S including k mutually exclusive latent subgroups $[S^1, S^2, \ldots, S^k]$, let e_i^t denote the ith entity in the tth subgroup of S. We state two criteria required for e_i^t to be typical in the entire set S:

Criterion 1 e_i^t should be representative in S^t.

Criterion 2 The significance of S^t in S should be high.

The typicality of e_i^t with respect to S is then defined as follows:

$$\text{Typ}(e_i^t, S) = L(e_i^t|S^t) \cdot \frac{|S^t|}{|S|} \tag{1}$$

where $L(e_i^t|S^t)$ measures the representativeness of e_i^t with regard to the subgroup S^t. In addition, $\frac{|S^t|}{|S|}$ indicates the relative size of S^t regarded as an estimator of significance. e_i^t is more typical when the number of entities in its subgroup is large.

The likelihood $L(e|S)$ of an entity e given a set of entities S is the posterior probability of e given S, which can be computed using probability density estimation methods. Many model estimation techniques have been proposed including parametric and nonparametric density estimations. We use kernel estimation [2] as it does not require any distribution assumption and can estimate unknown data distributions effectively. Kernel estimation is a generalization of sampling [14]. Moreover, we choose the commonly used Gaussian kernels. An illustration of the Gaussian kernel estimator is given in Fig. 2.

We set the bandwidth of the Gaussian kernel estimator $h = \frac{1.06s}{\sqrt[5]{n}}$ as suggested in [36], where n is the size of the data and s is the standard deviation of the data set. Formally, given a set of entities $S = (e_1, e_2, \ldots, e_n)$, the underlying likelihood function is approximated as:

$$L(e|S) = \frac{1}{n}\sum_{i=1}^{n} G_h(e, e_i) = \frac{1}{n\sqrt{2\pi}}\sum_{i=1}^{n} e^{-\frac{d(e,e_i)^2}{2h^2}} \tag{2}$$

where $d(e, e_i)$ is the cosine distance between e and e_i and $G_h(e, e_i)$ is a *Gaussian kernel*.

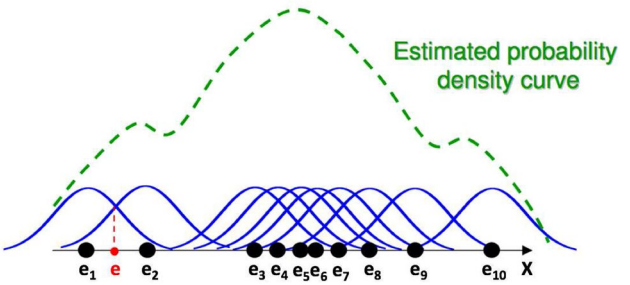

Fig. 2 Illustration of the Gaussian kernel estimator. An observation of the sample entity (e.g., e_1) increases the chance of observing other similar entities (e.g., e) nearby. Therefore, kernel estimation distributes the weight of each entity $\{e_1, e_2, \cdots e_{10}\}$ in the nearby space around according to a kernel function. Gaussian kernel estimator smooths out the contribution of each observed entity over a local neighborhood of that entity by Gaussian basis functions

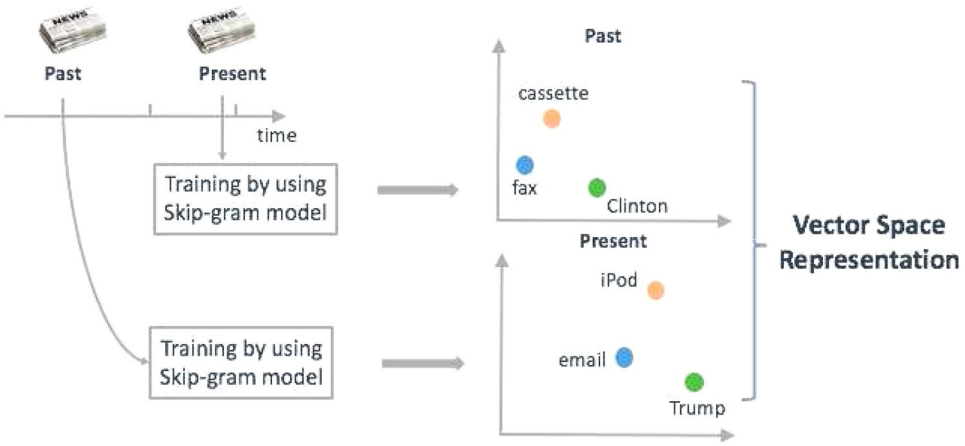

Fig. 3 Conceptual view of the across-time context alignment problem. If entity embeddings are learned independently for each period, most cost functions for training, such as skip-gram model, are invariant to rotations, as a by-product, the learned embeddings across time may not be placed in the same latent space

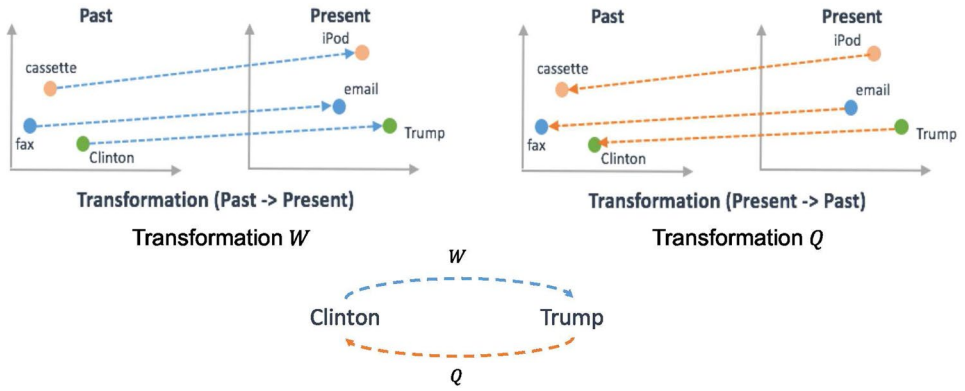

Fig. 4 Conceptual view of the bidirectional across-time transformation. For example, when we map *Clinton* from past into *Trump* in present, we should be able to map it back into past and obtain the original vector of *Clinton*. It can be then inferred that the transformation should be orthogonal

4 Measurement of Temporal Comparability

In this section, we describe the method for measuring temporal comparability between an entity e_A in set D_A and an entity e_B in the other set D_B. Intuitively, if e_A and e_B are comparable to each other, then e_A and e_B contain comparable aspects. For instance, (*iPod*, *Walkman*) could be regarded as comparable based on the observation that *Walkman* played the role of a popular portable music player 30 years ago same as *iPod* does nowadays. The key difficulty comes from the fact that there is low overlap between terms' contexts across time (e.g., the set of top co-occurring words with *iPod* in documents published in 2010s has typically little overlap with the set of top co-occurring words with *walkman* that are extracted from documents in 1980s). It is impossible to directly compare entities in two different semantic vector spaces, as the features in both spaces have no direct correspondence between each other (as can be seen in Fig. 3).Thus, our task is then to build the connection between semantic spaces of D_A and D_B.

Let transformation matrix W map the words from D_A into D_B, and transformation matrix Q map the words in D_B back into D_A. Let a and b be normalized word vectors from the news document collections D_A and D_B, respectively. The correspondence between words a and b can be evaluated as the similarity between vectors b and Wa, i.e., $\mathrm{Corr}(a,b) = b^{\mathrm{T}}Wa$. However, we could also form this correspondence as $\mathrm{Corr}'(a,b) = a^{\mathrm{T}}Qb$.

Theorem 1 *The linear transformations W and Q between spaces D_A and D_B are orthogonal.*

Proof To be self-consistent, we require $\mathrm{Corr}(a,b) = \mathrm{Corr}'(a,b)$, thus $b^{\mathrm{T}}Wa = (b^{\mathrm{T}}Wa)^{\mathrm{T}} = a^{\mathrm{T}}W^{\mathrm{T}}b = a^{\mathrm{T}}Qb$, and therefore $Q = W^{\mathrm{T}}$. Furthermore, when we map a term from D_A into D_B (as can be seen in Fig. 4), we should be able to map it back into D_A and obtain the original vector; hence, $a = Q(Wa) = W^{\mathrm{T}}(Wa)$. This expression should hold for any term in D_A, and we conclude that the transformation W should be an orthogonal matrix satisfying $W^{\mathrm{T}}W = I$ where I denotes the identity matrix. Thus, transformations W and its transpose Q are orthogonal. □

This observation has also been reported in [37, 42] for the purpose of bilingual text

Table 1 Examples of used common frequent terms

One, new, two, like, people,
Year, many, women, time, company,
Work, city, water, make, way,
Use, world, business, school, life

translation. Note that orthogonal transformation preserves vector norms, so given normalized vectors a and b, $\text{Corr}(a,b) = b^{\text{T}}Wa = |b||Wa|\cos(b,Wa) = \cos(b,Wa) = \text{Sim}_{\text{cosine}}(a,b)$.

However, the following challenge is that the training term pairs for learning the mapping W are difficult to obtain. We adopt here a trick proposed by [44] for preparing enough training data. Namely, we use so-called common frequent terms (CFTs) as the training term pairs. CFT is very frequent terms in both compared document collections (see Table 1 for some examples of used CFTs in this work). Such frequent terms tend to change their meanings only to a small extent across time. The phenomenon that words which are intensively used in everyday life evolve more slowly has been reported in several languages including English, Spanish, Russian and Greek [13, 23, 29].[1]

Given L pairs composed of normalized vectors of CFTs trained in both document collections $[(a_1,b_1), (a_2,b_2), \cdots, (a_L,b_L)]$, we should learn the transformation W by maximizing the accumulated cosine similarity of CFT pairs (we utilize the top 5% (18k words) of common frequent terms to train the transformation matrix),

$$\max_{W} \sum_{i=1}^{L} b_i^{\text{T}} W a_i, \quad \text{s.t. } W^{\text{T}}W = I \tag{3}$$

To infer the orthogonal transformation W from pairs of CFTs $\{a_i, b_i\}_{i=1}^{L}$, we state the following theorem.

Theorem 2 *Let A and B denote two matrices, such that the ith row of (A, B) corresponds to pair of vectors $(a_i^{\text{T}}, b_i^{\text{T}})$. By computing the SVD of $M = A^{\text{T}}B = U\Sigma V^{\text{T}}$, the optimized transformation matrix W^* satisfies*

$$W^* = U \cdot V^{\text{T}} \tag{4}$$

Proof Maximizing $\sum_{i=1}^{L} b_i^{\text{T}} W a_i$ equals to maximizing $tr(BWA^{\text{T}}) = tr(A^{\text{T}}BW) = tr(U\Sigma V^{\text{T}}W) = tr(\Sigma V^{\text{T}}WU)$. Let $Z = V^{\text{T}}WU$, for Z is orthogonal (being the product of orthogonal matrices), thus $\sum_j Z_{i,j}^2 = 1$ and $Z_{i,i} \leq 1$. Then, $tr(\Sigma V^{\text{T}}WU) = tr(\Sigma Z) = \sum_i \Sigma_{i,i} Z_{i,i} \leq \sum_i \Sigma_{i,i}$. The last inequality holds because $\Sigma_{i,i} \geq 0$ (given Σ is obtained by SVD). Then, the objective can achieve the maximum if $Z_{i,i} = 1$ which implies $Z = I$. So an optimal W^* is UV^{T}. □

The obtained orthogonal transformation W^* allows for learning correspondence on the level of terms. Based on it, we measure the temporal comparability between an entity e_A in set D_A and an entity e_B in the other set D_B as follows:

$$\text{Comp}(e_A, e_B) = \text{Sim}_{\text{cosine}}(W^* \cdot e_A, e_B) \tag{5}$$

5 ILP Formulation for Detecting Comparables

In this section, we describe our method for discovering comparable entity pairs. Given two sets of entities D_A and D_B the output is m comparable entity pairs $[p_1, p_2, \cdots, p_m]$, where each pair contains an entity from D_A and an entity from D_B. Inspired by AP algorithm [11], we formulate our task as a process of identifying a subset of typical comparable entity pairs. It has been empirically found that using AP for solving objectives such as in our case (see Eq. 6) suffers considerably from convergence issues [43]. Thus, we propose a concise integer linear programming (ILP) formulation for discovering comparable entities, and we use the *branch-and-bound* method to obtain the optimal solution.

The prior here is that the ratio of typical candidate entities over trivial candidate entities is very low given the output length limit, and the input entity sets can be very diverse. We thus call an entity as exemplar if it represents a latent subgroup and is voted by other entities. Intuitively, if the selected exemplars concisely represent the entire entity set, their typicality and diversity will naturally arise. We then formulate the task as a process of selecting a subset of k_A and k_B exemplars for each set, respectively, and choosing m entity pairs based on the identified exemplars. Each non-exemplar entity is assigned to an exemplar entity based on a measure of similarity, and each exemplar e represents a subgroup comprised of all non-exemplar entities that are assigned to e. On the one hand, we wish to maximize the overall typicality of selected exemplars w.r.t. their representing subgroups. On the other hand, we expect to maximize the overall comparability of the top m entity pairs, where each pair consists of two exemplars from different sets.

We next introduce some notations used in our method. Let e_i^A denote the ith entity in D_A. $M_A = [m_{ij}]^A$ is a $n_A \times n_A$ binary square matrix such that n_A is the number of entities within D_A. m_{ii}^A indicates whether entity e_i^A is selected as an exemplar or not, and $m_{ij:i\neq j}^A$ represents whether entity e_i^A votes for entity e_j^A as its exemplar. Similar to M_A, the $n_B \times n_B$ binary square matrix M_B indicates how entities belonging to D_B choose their

[1] Note that even if certain CFTs do not retain the same semantics across time, the results should not deteriorate significantly when the number of used CFTs is sufficiently high.

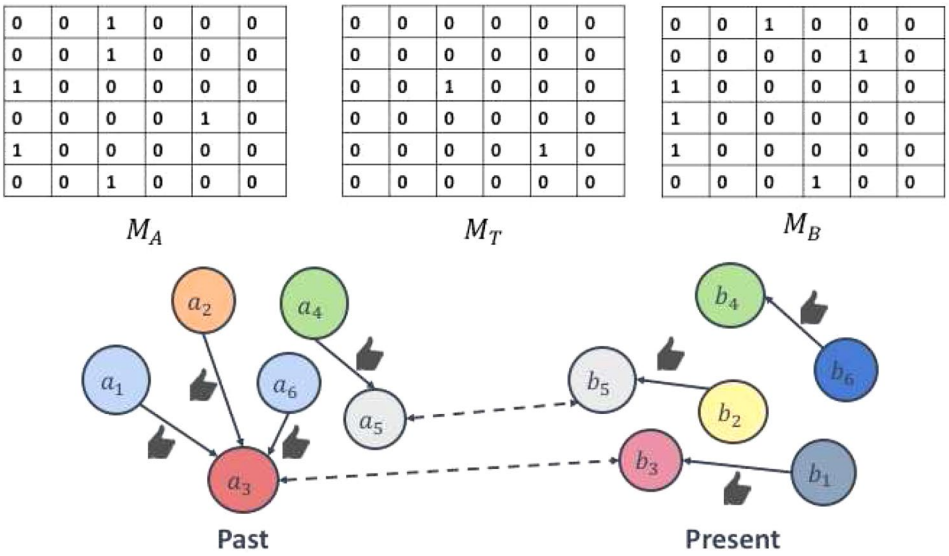

Fig. 5 An toy example of our exemplar identification task and the corresponding matrices M_A, M_B and M_T. An entity is regarded as an exemplar if it represents a latent subgroup and is voted by other entities. In this example, $\{a_3, a_5\}$ are selected exemplars representing the past, and $\{b_3, b_5\}$ are identified exemplars of the present time. Based on them, (a_3, b_3) and (a_5, b_5) are chosen exemplar pairs

exemplars, where n_B is the number of entities within D_B. m_{ii}^B indicates whether entity e_i^B is selected as an exemplar or not, and $m_{ij:i\neq j}^B$ represents whether entity e_i^B votes for entity e_j^B as its exemplar. Different from M_A and M_B, $M_T = [m_{ij}]^{\mathrm{T}}$ is a $n_A \times n_B$ binary matrix whose entry m_{ij}^{T} denotes whether entities e_i^A and e_j^B are paired together as the final result. Figure 5 shows an example of the exemplar identification task and the corresponding matrices M_A, M_B and M_T. Then, the following ILP problem is designed for the task of selecting k_A and k_B exemplars for each set, respectively, and for selecting m comparable entity pairs:

$$\max\ \lambda \cdot m \cdot \left[T'(M_A) + T'(M_B)\right] + (1-\lambda)\cdot(k_A + k_B)\cdot C'(M_T) \tag{6}$$

$$T'(M_X) = \sum_{i=1}^{n_X} m_{ii}^X \cdot \mathrm{Typ}(e_i^X, G(e_i^X)), X \in \{A, B\} \tag{7}$$

$$C'(M_T) = \sum_{i=1}^{n_A}\sum_{j=1}^{n_B} m_{ij}^{\mathrm{T}} \cdot \mathrm{Comp}(e_i^A, e_j^B) \tag{8}$$

$$G(e_i^X) = \left\{ e_j^X | m_{ji}^X = 1, j \in \{1, \ldots, n_X\} \right\}, \quad i \in \{1, \ldots, n_X\}, X \in \{A, B\} \tag{9}$$

$$\text{s.t.}\quad m_{ij}^X \in \{0,1\}, i \in \{1, \ldots, n_X\}, \quad j \in \{1, \ldots, n_X\}, X \in \{A, B\} \tag{10}$$

$$m_{ij}^{\mathrm{T}} \in \{0,1\}, \quad i \in \{1, \ldots, n_A\}, j \in \{1, \ldots, n_B\} \tag{11}$$

$$\sum_{i=1}^{n_X} m_{ii}^X = k_X, \quad X \in \{A, B\} \tag{12}$$

$$\sum_{j=1}^{n_X} m_{ij}^X = 1, \quad i \in \{1, \ldots, n_X\}, X \in \{A, B\} \tag{13}$$

$$m_{jj}^X - m_{ij}^X \geq 0, \quad i \in \{1, \ldots, n_X\}, j \in \{1, \ldots, n_X\}, X \in \{A, B\} \tag{14}$$

$$\sum_{i=1}^{n_A}\sum_{j=1}^{n_B} m_{ij}^{\mathrm{T}} = m \tag{15}$$

$$m_{ii}^A - m_{ij}^{\mathrm{T}} \geq 0, \quad i \in \{1, \ldots, n_A\}, j \in \{1, \ldots, n_B\} \tag{16}$$

$$m_{jj}^B - m_{ij}^{\mathrm{T}} \geq 0, \quad i \in \{1, \ldots, n_A\}, j \in \{1, \ldots, n_B\} \tag{17}$$

$$\sum_{j=1}^{n_B} m_{ij}^{\mathrm{T}} \leq 1, \quad i \in \{1, \ldots, n_A\} \tag{18}$$

$$\sum_{i=1}^{n_A} m_{ij}^{\mathrm{T}} \leq 1, \quad j \in \{1, \ldots, n_B\} \tag{19}$$

We now explain the meaning of the above formulas. First, Eq. (12) forces that k_A and k_B exemplars are identified for both sets D_A and D_B, respectively, and Eq. (15) guarantees that m entity pairs are selected as the final result. The restriction given by Eq. (13) means each entity must choose only

one exemplar. Equation (14) enforces that if one entity e_j^X is voted by at least one other entity, then it must be an exemplar (i.e., $m_{jj}^X = 1$). The constraint given by (16) and (17) jointly guarantees that if an entity is selected in any comparable entity pair (i.e., $m_{ij}^{\mathrm{T}} = 1$), then it must be an exemplar in its own subgroup (i.e., $m_{ii}^A = 1$ and $m_{jj}^B = 1$). Restricted by Eqs. (18) and (19), each selected exemplar in the result is only allowed to appear once to avoid redundancy. $T'(M_X)$ represents the overall typicality of selected exemplars in both sets D_A and D_B, and $G(e_i^X)$ denotes the representing subgroup for entity e_i^X (if e_i^X is not chosen as an exemplar, its representing subgroup will be null). $C'(M_T)$ denotes the overall comparability of generated entity pairs. In view of the fact that there are $(k_A + k_B)$ values (each value is in [0,1]) in the typicality component $T'(M_A) + T'(M_B)$, and m numbers (each number is in [0,1]) in the comparability part $C'(M_T)$, we add the coefficients m and $(k_A + k_B)$ in the objective function to avoid suffering from skewness problem. Finally, the parameter λ[2] is used to balance the weight of the two parts. Our proposed ILP formulation guarantees to achieve the optimal solution by using *branch-and-bound method*.

6 Experiments

6.1 Datasets

We perform the experiments on the New York Times Annotated Corpus [34]. The dataset is publicly available at the specified URL.[3] This corpus is a collection of 1.8 million articles published by the New York Times between January 01, 1987, and June 19, 2007, and has been frequently used to evaluate different researches that focus on temporal information processing or extraction in document archives [3]. For the experiments, we divide the corpus into four parts based on article publication dates: [1987, 1991],[1992, 1996], [1997, 2001] and [2002, 2007]. The vocabulary size of each time period is around 300k. We first set on comparing the pair of time periods which are separated by the longest time gap, [1987, 1991] (denoted as T_{A1}) and [2002, 2007] (denoted as T_B). We denote this comparison as C_1. We assume here that the more the two time periods are farther apart, the stronger is the context change, which increases the difficulty of finding corresponding entity pairs. We then perform additional experiment using another past time period [1992, 1996] (denoted as T_{A2}) to verify whether our approach is still superior on different past time. This comparison is denoted as C_2.

We obtain the distributed vector representations for time periods T_{A1}, T_{A2} and T_B by training the skip-gram model using the gensim Python library [32]. The number of dimensions of word vectors is set to be 200, following previous work [30] which has experimented on the embedding dimension for the analogy task.

To prepare the entity sets for each period, we retain all unigrams and bigrams which appear more than 10 times in the collection of news articles within that period, excluding stopwords and all numbers. We then adopt spaCy[4] for recognizing named entities based on all unigrams and bigrams. The details of identified entities are shown in Table 2. The meaning of sub-categories can be found at spaCy Web site.[5] Note that some sub-categories of entities were not used due to their weak significance, e.g., TIME/DATE.

6.2 Test Sets

As far as we know, there is no ground truth data available for the task of identification of across-time comparable entities. Hence, we then apply pooling technique for creating test sets. In particular, we have leveraged the pooling technique by pulling the resulting comparable entity pairs from all the proposed methods and baselines as listed in Sect. 6.4). Three annotators then judged every result in the pool based on the following steps: firstly highlight all the typical entities in the results, and then create reference entity pairs based on the highlighted entities. There was no limit on the number of highlighted entities nor chosen entity pairs. The annotators did not know which systems generated which answers. They were allowed to utilize any external resources[6] or use search engines in order to verify the correctness of the results. In total, 447 entities, 342 entities and 315 entities were chosen as typical exemplars for periods T_{A1}, T_{A2} and T_B, respectively. Among them, 168 pairs and 109 pairs were constructed.

6.3 Evaluation Criteria

6.3.1 Criteria for Quantitative Evaluation

Given the human-labeled typical entity set and the comparable entity pairs' set, we compare the generated results with the ground truth. We compute *precision*, *recall* and F_1-*score* to measure the performance of each method, respectively, as follows:

$$\text{Precision} = \frac{\{\text{Result\_pairs}\} \cap \{\text{labeled\_pairs}\}}{\{\text{result\_pairs}\}} \tag{20}$$

[2] We experimentally set the value of λ to be 0.4 in Sect 6.

[3] https://catalog.ldc.upenn.edu/LDC2008T19

[4] https://github.com/explosion/spaCy.

[5] https://spacy.io/api/annotation#named-entities.

[6] There are several online available test sets for checking across-time term correspondence, such as the one at https://github.com/yifan0sun/DynamicWord2Vec.

Table 2 Summary of datasets

Period	LOC	PRODUCT	NORP	WOA	GPE	PERSON	FACT	ORG
T_{A1}	427	87	2959	129	7810	33,127	328	23,775
T_{A2}	370	57	2203	103	4698	27,932	247	16,751
T_B	304	44	1573	91	4460	16,103	221	11,215
Period	LAW	EVENT	TOTAL					
T_{A1}	18	212	68,872					
T_{A2}	10	149	52,520					
T_B	11	129	34,151					

$$\text{Recall} = \frac{\{\text{Result\_pairs}\} \cap \{\text{labeled\_pairs}\}}{\{\text{labeled\_pairs}\}} \tag{21}$$

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{recall}}{\text{Precision} + \text{recall}} \tag{22}$$

6.3.2 Criteria for Qualitative Evaluation

To further evaluate the quality of the results, we also conducted user-based analysis. In particular, three subjects were invited to annotate the results generated by each method using the following quality criteria: (1) *Correctness*—it measures how sound the results are. (2) *Comprehensibility*—it measures how easy it is to understand and explain the results. (3) *Diversity*—it quantifies how varying and diverse information the annotators could acquire. All the scores were given in the range from 1 to 5 (1: not at all, 2: rather not, 3: so so, 4: rather yes, 5: definitely yes). We averaged all the individual scores given by the annotators to obtain the final scores per each comparison. During the assessment, the annotators were allowed to utilize any external resources including the Wikipedia, Web search engines, books, etc.

6.4 Baselines

We prepare different methods to select temporally comparable entity pairs. We first compare our model with three widely used clustering methods: k-means clustering, DBSCAN clustering [7] and aforementioned AP clustering [11]. Besides, we also adopt the mutually reinforced random walk model [4] (denoted as MRRW) to judge entity typicality based on the hypothesis that typical exemplars are those who are similar to the other members of its category and dissimilar to members of the contrast categories. Finally, we also test a limited version of our approach called independent ILP (denoted as I-ILP) that separately identifies exemplars of each input sets based on our proposed ILP framework. I-ILP aims to maximize the overall typicality of selected exemplars for each set, respectively, without considering whether chosen exemplars are comparable or not. In this study, we use the Gurobi solver [12] for solving the proposed ILP framework. After the exemplars have been selected by the above methods, we construct the entity pairs which have the maximal comparability based on identified exemplars as follows.

$$\text{P} \equiv \ \text{argmax} \sum_{i=1}^{m} Comp(e_i^A, e_i^B) \tag{23}$$

where $P = [p_1, p_2, ..., p_m]$ are expected comparables and $p_i = (e_i^A, e_i^B)$. e_i^A and e_i^B are chosen exemplars from the compared sets.

Besides, we also test effectiveness of orthogonal transformation for computing across-time comparability. To this end, we test the method which directly compares the vectors trained in different time periods separately without any transformation (denoted as Embedding-S + Non-Tran). Moreover, we also analyze the methods which utilize the distributional entity representation trained on the combination of news articles from two compared periods jointly (denoted as Embedding-J). We denote the proposed transformation-based methods as Embedding-S + OT.

6.5 Experiment Settings

We set the parameters as follows:

(1) *Number of subgroups of each input set* Following [40], we set the number k of latent subgroups of each input set as:

$$k = \lceil \sqrt{n} \rceil \tag{24}$$

where n is the number of entities in the set.

(2) *Number of generated pairs for comparison* In view of the fact that the number of counterparts for each entity is at most one in the output, we set the number of generated pairs m to be its lower bound $\min\{k_A, k_B\}$, where k_A and k_B are the numbers of identified exemplars of two compared entity sets.

6.6 Evaluation Results

6.6.1 Results of Quantitative Evaluation

Tables 3 and 4 show the performance of all the analyzed methods in terms of *precision*, *recall* and F_1-*score* for comparison C_1 and C_2, respectively, while we show the detailed results for a few examples in Table 5. We first notice that the performance is extremely poor without transforming the contexts of entities. Only very few results in *Non-Tran* approaches are judged as correct. On the other hand, although methods based on the jointly trained word embeddings perform better than *Non-Tran*, the performance increase is quite limited. It can be observed that the across-time orthogonal transformation is quite helpful since it exhibits significantly better effectiveness in terms of all the metrics than the other two types of methods. This observation suggests little overlap in the contexts of news articles separated by longer time gaps and that the task of identifying temporal analogous entities is quite difficult.

Moreover, a closer look at Tables 3 and 4 reveals that regardless of the type of evaluation metric, J-ILP improves the performance of the other models under transformation. From Tables 3 and 4, it can be seen that 27.3% and 17.8% entity pairs generated by J-ILP model are judged as correct by human annotators and that 29.0% and 30.3% of ground truth entity pairs are discovered, for C_1 and C_2, respectively. Specifically, J-ILP improves the baselines by 87.3% and 30.2% when measured using the main metric F_1-*score* on average, respectively. These results are observed because the proposed J-ILP formulation takes both necessary factors (typicality and comparability) into consideration. Based on this formulation, the optimal solution can be obtained using the *branch-and-bound* method.

We also investigate the possible reasons for the poor performance of baselines. K-means suffers from strong sensitivity to outliers and noise, which leads to a varying performance. On the other hand, although AP shares many similar characteristics with J-ILP, its belief propagation mechanism does not guarantee to find the optimal solution, hence its lower performance. DBSCAN relies on the concept of "core point" for identifying exemplars with high density; however, it is possible that a typical point does not have many points lying close to it, and a "core point" may not be typical in the scenarios of unbalanced clusters. Finally, MRRW tends to select entities that contain more discriminative features rather than common traits, which can explain why it has worse performance.

6.6.2 Results of Qualitative Evaluation

Figures 6 and 7 show the evaluation scores in terms of *Correctness*, *Comprehensibility* and *Diversity* judged by annotators, for C_1 and C_2, respectively. We first note that our J-ILP model achieves better results than the baselines based on both *Correctness* and *Comprehensibility* criteria. On average, J-ILP outperforms baselines by 20.2% and 28.0% in terms of *Correctness* and *Comprehensibility* for comparison C_1, and by 18.6% and 23.0% for C_2, respectively. This observation proves that J-ILP has relatively good performance in detecting

Table 3 Performance of models in terms of *precision*, *recall* and F_1-*score* comparing periods [1987, 1991] and [2002, 2007] (comparison C_1). The best results of each setting are indicated in bold, while the best overall results are underlined

Method	Embedding-S+Non-Tran			Embedding-J			Embedding-S+OT		
	Prec.	Recall	F_1-score	Prec.	Recall	F_1-score	Prec.	Recall	F_1-score
K-means	**0.027**	**0.030**	**0.028**	0.081	0.089	0.085	0.186	0.195	0.190
DBSCAN	0.000	0.000	0.000	0.000	0.000	0.000	0.105	0.106	0.105
AP	0.016	0.018	0.017	0.049	0.054	0.051	0.154	0.160	0.156
MRRW	0.000	0.000	0.000	0.027	0.030	0.028	0.132	0.136	0.133
I-ILP	0.016	0.018	0.017	0.049	0.054	0.051	0.165	0.171	0.167
J-ILP	0.000	0.000	0.000	**0.124**	**0.137**	**0.130**	**0.273**	**0.290**	**0.281**

Table 4 Performance of models in terms of *precision*, *recall* and F_1-*score* comparing periods [1992, 1996] and [2002, 2007] (comparison C_2). The best results of each setting are indicated in bold, while the best overall results are underlined

Method	Embedding-S+Non-Tran			Embedding-J			Embedding-S+OT		
	Prec.	Recall	F_1-score	Prec.	Recall	F_1-score	Prec.	Recall	F_1-score
K-means	0.005	0.009	0.006	0.038	0.064	0.048	0.146	0.248	0.184
DBSCAN	0.000	0.000	0.000	0.027	0.046	0.034	0.092	0.156	0.116
AP	0.000	0.000	0.000	0.016	0.028	0.020	0.141	0.239	0.177
MRRW	0.000	0.000	0.000	0.016	0.028	0.020	0.157	0.266	0.197
I-ILP	**0.011**	**0.018**	**0.017**	0.043	0.073	0.054	0.151	0.257	0.190
J-ILP	0.000	0.000	0.000	**0.059**	**0.101**	**0.075**	**0.178**	**0.303**	**0.224**

Table 5 Example results where entity pairs are ground truth

Entity pair	K-means	DBSCAN	AP	MRRW	I-ILP	J-ILP
(iraq, syria)	(0,0)	(1,1)*	(1,0)	(1,0)	(1,1)*	(1,1)
(president_reagan, george_bush)	(1,1)*	(0,1)	(0,1)	(0,1)	(1,0)	(1,1)
(american_express, credit_card)	(1,0)	(0,0)	(0,0)	(0,0)	(1,1)	(1,1)
(macintosh, pc)	(1,1)	(1,0)	(0,0)	(0,0)	(1,0)	(1,0)
(salomon, morgan_stanley)	(0,1)	(0,0)	(1,0)	(1,0)	(0,1)	(1,1)
(national_basketball, world_series)	(1,1)	(0,0)	(0,1)	(0,0)	(0,1)	(0,1)
(european_community, china)	(0,1)	(0,0)	(0,1)	(1,0)	(0,0)	(0,1)
(pan_am, american_airlines)	(1,1)*	(1,0)	(1,1)*	(0,0)	(1,1)	(1,0)
(mario_cuomo, george_pataki)	(0,1)	(1,0)	(0,1)	(0,0)	(1,1)*	(1,1)
(bonn, berlin)	(0,0)	(0,0)	(1,0)	(1,0)	(1,1)	(1,1)
(sampras, federer)	(0,0)	(1,1)	(0,0)	(0,0)	(0,1)	(0,0)
(saddam, al_qaeda)	(1,1)	(1,0)	(1,0)	(0,1)	(0,0)	(1,0)

The entity on the left in parentheses is from period [1987, 1991] while the entity on the right is from [2002, 2007]. The tags (0, 1) shown in parentheses denote the appearance of ground truth entity in results (1 means the entity matches the ground truth exemplars, while 0 means otherwise). Note that only the tag (1,1) indicates the ground truth entity pair was identified correctly, while (1,1)* denotes that although both entities are recognized as exemplars, they are not paired together in the results

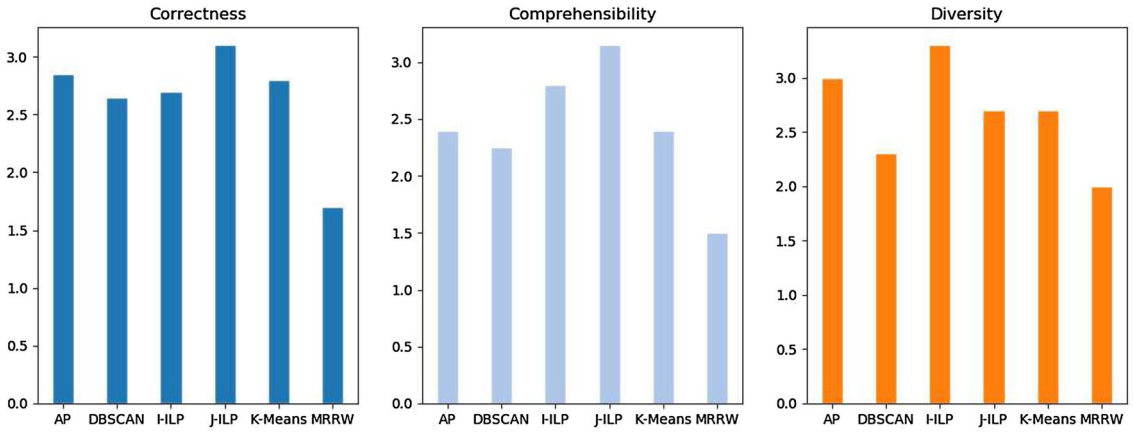

Fig. 6 Performance of models in terms of *Correctness*, *Comprehensibility* and *Diversity* comparing periods [1987, 1991] and [2002, 2007] (comparison C_1)

dominant and reasonable entity pairs, which tend to be highly scored by annotators. On the other hand, J-ILP underperforms two baselines AP algorithm and I-ILP in terms of diversity. It may be because AP algorithm and I-ILP are intrinsically better in capturing representative and diverse exemplars, while J-ILP aims to balance the entity typicality and comparability simultaneously.

6.7 Additional Observations

6.7.1 Additional Metrics

We additionally examine the quality of generated entity pairs P (where $P = [p_1, p_2, \cdots, p_m]$, and $p_i = (e_i^A, e_i^B)$) by the following metrics:

Typicality (Typ) measures the representativeness of chosen entities, where $Typ(e_i^A)$ and $Typ(e_i^B)$ are computed using Eq. (1).

$$\text{Typ}(P) = \sum_{i=1}^{m} (\text{Typ}(e_i^A) + \text{Typ}(e_i^B)) \tag{25}$$

Comparability (Comp) is expressed as

$$\text{Comp}(P) = \sum_{i=1}^{m} \text{Sim}_{\text{cosine}}(e_i^A, e_i^B) \tag{26}$$

Product of typicality and comparability (product) which takes into account both Typ and Comp, thus reflecting the overall quality of entity pairs.

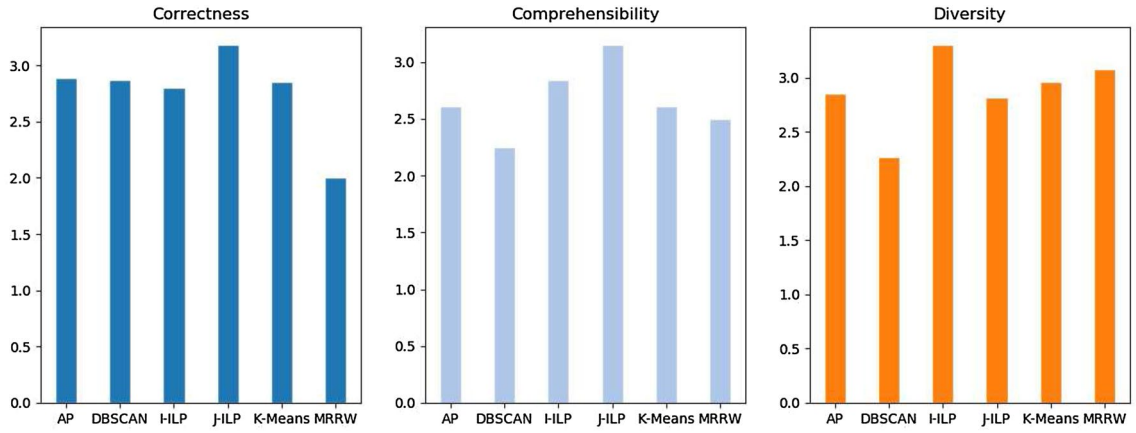

Fig. 7 Performance of models in terms of *Correctness*, *Comprehensibility* and *Diversity* comparing periods [1992, 1996] and [2002, 2007] (comparison C_2)

Table 6 Performance of models over all datasets in terms of *typicality*, *comparability* and *product*. The best result of each setting is indicated in bold

Model	Embedding-S+Non-Tran			Embedding-J			Embedding-S+OT		
	Typ	Comp	Product	Typ	Comp	Product	Typ	Comp	Product
K-means	50.505	2.361	119.242	57.470	2.680	154.020	62.052	3.263	202.476
DBSCAN	49.504	2.320	114.849	56.862	3.495	198.733	61.092	3.786	231.294
AP	50.573	2.333	117.987	57.635	3.105	178.957	61.743	3.562	219.929
MRRW	46.043	2.069	95.263	48.136	2.348	113.023	56.588	3.534	199.982
I-ILP	**51.037**	2.167	110.597	**58.592**	2.680	157.027	**62.799**	3.339	209.686
J-ILP	49.467	**3.186**	**157.602**	56.207	**3.712**	**208.640**	60.752	**3.903**	**237.115**

$$\text{Product}(P) = \text{Typ}(P) \cdot \text{Comp}(P) \tag{27}$$

Table 6 shows the performance in terms of *typicality*, *comparability* and *product*. Again, it can be obviously observed that the performance of all analyzed methods is poor when the context of entities across time is not transformed or mixed together. The orthogonal transformation shows significant effectiveness in aligning the contexts of news articles separated by long time gaps.

In addition, we discover that J-ILP has significantly better performance than all the baseline methods in terms of two main evaluation metrics, *comparability* and *product*. Not surprisingly, we notice that I-ILP outperforms the other methods in terms of *typicality*, for it was intrinsically designed to achieve the maximal exemplar typicality without considering comparability among exemplars. More specifically, J-ILP outperforms baselines by 27.3% and 27.6% in terms of *comparability* and *product*, respectively. Such observation implies that the superior performance of J-ILP on generating proper typical temporal comparables can be mainly due to its high sensitivity to information "shared" by different periods.

6.7.2 Effects of the Number of CFTs

We examine now how the performance of orthogonal mapping varies when we change the rate of used CFTs to train the transformation matrix. We test the rate within the range [0, 0.1] and with a step of 0.01. Figure 8 shows the *precision*, *recall* and F_1-*score* curves of our method, respectively. We can see from the figure that the rate of used CFTs has an effect on the performance of mapping. In this study, we set the rate of CFTs as 0.05 based on the observations received from Fig. 8 to obtain the best results.

6.7.3 Effects of Trade-Off Parameter

We perform a grid search to find the best trade-off parameter λ. We set λ in the range [0.0, 1.0] with a step of 0.1. Note that when $\lambda = 1.0$, the J-ILP formulation degenerates into the aforementioned I-ILP model. From Fig. 9, we see that when λ is within the range [0.0, 0.4], the performance of J-ILP reaches its maximal value and remains stable. On the other hand, the values of all metrics degrade when increasing the value of λ after $\lambda = 0.4$. In general, we can see that λ needs to be fine-tuned to achieve an optimal performance. In

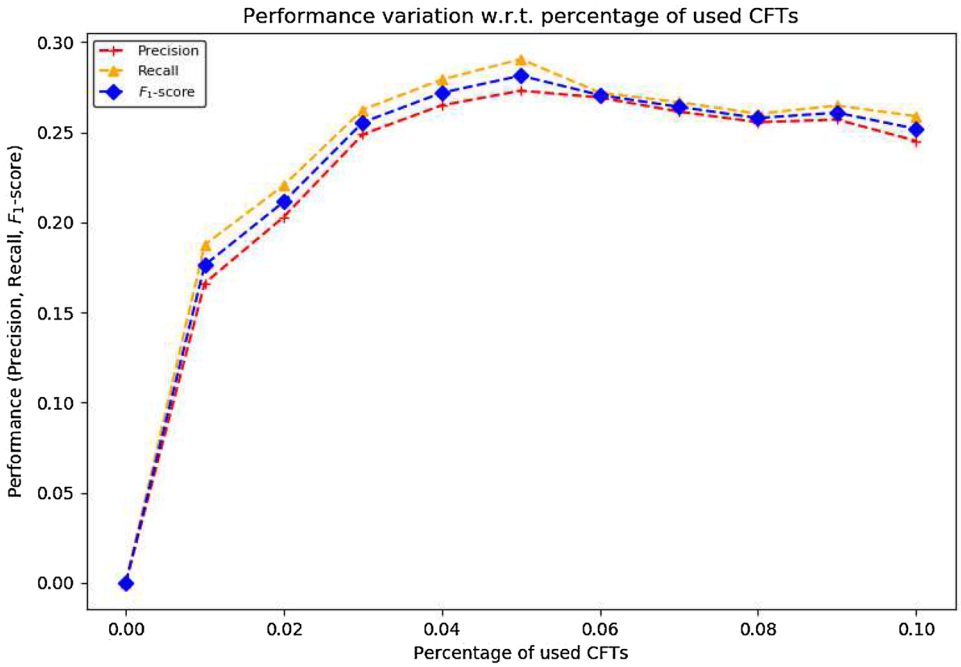

Fig. 8 Performance variation of *precision*, *recall* and F_1-*score* w.r.t. percentage of used CFTs. (see Sect. 6.7.2)

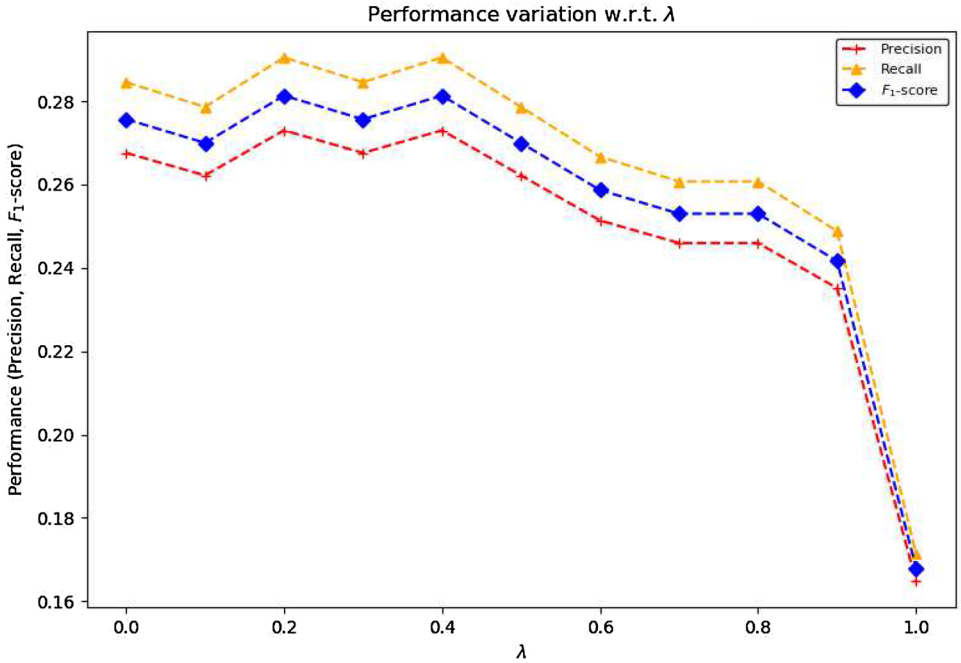

Fig. 9 Performance variation of *precision*, *recall* and F_1-*score* w.r.t. λ. (see Sect. 6.7.3)

this study, we set λ as 0.4 based on the observations received from Fig. 9.

6.7.4 Sensitivity to Kernel Choice

In this work, we adopt Gaussian kernel function for computing entity typicality. Let the generated pairs returned by using Gaussian kernel be P_G and the results generated by other popular kernel functions be P_O. The difference of P_G and P_O is measured as the difference rate d as follows.

$$d = \frac{|P_G - PO|}{|PG|} \cdot 100\% \tag{28}$$

Table 7 shows that the exemplars identified by different kernels are in general consistent, as the difference rate d is low.

Table 7 Difference rate versus kernel function

Kernel function	Quatic	Triweight	Epanechnikov	Cosine
Difference rate	15.9	10.3	5.5	15.9

7 Related Work

Comparable Entity Mining The task of comparable entity mining has attracted much attention in the NLP and Web mining communities [15–19, 22]. Approaches to this task include hand-crafted extraction rules [8], supervised machine-learning methods [26, 35] and weakly supervised methods [17, 22]. Jindal et al. [18, 19] were the first to propose a two-step system in finding comparable entities which first tackles a classification problem (i.e., whether a sentence is comparative) and then a labeling problem (i.e., which part of the sentence is the desideratum). Later work refined that system by using a bootstrapping algorithm [22], or extended the idea of mining comparables to different types of corpora including query logs [16, 17] and comparative questions [22]. In addition, comparable entity mining is strongly related to the problem of automatic structured information extraction, comparative summarization and named entity recognition. Some work lies in the intersection of these tasks [10, 24].

Temporal Analog Detection and Embeddings Alignment A part of our system approaches the task of identifying temporally corresponding terms across different times. The related work to this subtask includes computing term similarity across time [1, 20, 21, 38]. In this study, we represent terms using the distributed vector representation [27]. Thus, the problem of connecting news articles' context across different time periods can be approached by aligning pretrained word embeddings in different time periods. Mikolov et al. proposed a linear transformation aligning bilingual word vectors for automatic text translation such as translation from Spanish to English [28]. Faruqui et al. obtained bilingual word vectors using CCA [9]. More recently, Xing et al. argued that the linear matrix adopted by Mikolov et al. should be orthogonal [42]. Similar suggestion has been given by Samuel et al. [37]. Besides linear models, nonlinear models such as "deep CCA" have also been introduced for the task of mapping multilingual word embeddings [25]. In this study, we adopt the orthogonal transformation for computing across-time entity correspondence due to its high accuracy and efficiency.

To the best of our knowledge, we are the first to focus on the task of automatically generating across-time comparable entity pairs given two entity sets, and on using the notion of typicality analysis from cognitive science and psychology.

8 Discussions

We discuss here several relevant aspects to the task of mapping entity sets in news archives across time.

- In the problem setting, the compared entity sets are extracted from two different time periods. However, our proposed J-ILP model also works for mapping two entity sets from the same time period (e.g., comparing European politicians with contemporary Asian politicians). Note that in this case, we do not need to solve the across-time context alignment problem. Entity vectors from different sets can be compared directly based on their cosine similarity.
- On the other hand, when mapping entity sets across time, the same entity which appears in both time periods may be discovered and paired together in the result, in case that such entity is associated with a stable diachronic meaning (e.g., *New York*, *dollar* etc.). However, entities with changed roles will less likely be included in the result, since the temporal comparability between their meaning at different times is low. In this task, we focus more on generating similar entity pairs that are beneficial for understanding the connection between two time periods.
- We show a few examples of mapped entity pairs in Table 5. For instance, (*president_reagan*, *george_bush*) and (*mario_cuomo*, *george_pataki*) represent the corresponding US President and the governor of New York in different periods, respectively. As another two examples, (*bonn*, *berlin*) represents the pair of German capital before and after the reunification of Western and Eastern German. (*salomon*, *morgan_stanley*) describes the pair of temporal corresponding large investment banks. We can see these detected pairs are clear and conveying comparative historical knowledge.
- Our framework relies on the orthogonal transformation for computing across-time entity comparability and a ILP framework for generating exemplar entity pairs. However, the transfer of deep learning framework to our task may bring new insights. For example, we may use an approach similar to the one proposed in [31] for discovering across-time analogy relationships.
- Finally, there are some connections between our research problems with the knowledge graph embedding task (KGE). A knowledge graph is a multi-relational graph composed of entities (nodes) and relations (different types of edges), and the goal of knowledge graph embedding is to embed components of a knowledge graph including entities and relations into continuous vector spaces, so as to simplify the manipulation while preserving the inherent structure of the knowl-

edge graph [41]. In our task, we also focus on the triple of the form (*entity from period* T_A, *temporal analog*, *entity from period* T_B), where *temporal analog* denotes the relation between entities from T_A and T_B. However, the difference lies in that we focus on the construction of such triples from news archives, while KGE aims to learn effective representations based on these triples.

9 Conclusions and Future Work

Entity comparison has been studied in the past (e.g., [33]). For an input entity pair, the task was to find their differences and similarities. Other work focused on finding a comparable entity for a given input entity [15, 17]. In this paper, we propose a novel entity-oriented comparison style over time in which entire subsets of entities from different time periods are matched to generate sets of comparable entity pairs. We introduce an approach that conducts such across-time comparison based on finding typical exemplars. Picking exemplars is an effective strategy used by humans for obtaining contrastive knowledge or for understanding unknown entity groups by their comparison to familiar groups (e.g., entities from the past compared to ones from present). We adopt a concise ILP model for maximizing the overall representativeness and comparability of the selected entity pairs. The experimental results demonstrate the effectiveness of our model compared to several competitive baselines.

In future, we will test our model on more heterogeneous and larger datasets where contexts of entities may be more difficult to be compared.

Acknowledgements This research has been supported by JSPS KAKENHI grants (#17H01828, #18K19841).

References

1. Berberich K, Bedathur SJ, Sozio M, Weikum G (2009) Bridging the terminology gap in web archive search. In: WebDB
2. Breiman L, Meisel W, Purcell E (1977) Variable kernel estimates of multivariate densities. Technometrics 19(2):135–144
3. Campos R, Dias G, Jorge AM, Jatowt A (2015) Survey of temporal information retrieval and related applications. ACM Comput Surv (CSUR) 47(2):15
4. Chen YN, Metze F (2012) Two-layer mutually reinforced random walk for improved multi-party meeting summarization. In: SLT, 2012 IEEE. IEEE, pp 461–466
5. Duan Y, Jatowt A, Bhowmick SS, Yoshikawa M (2019) Typicality-based across-time mapping of entity sets in document archives. In: International conference on database systems for advanced applications. Springer, pp 350–366
6. Dubois D, Prade H, Rossazza JP (1991) Vagueness, typicality, and uncertainty in class hierarchies. Int J Intell Syst 6(2):167–183
7. Ester M, Kriegel HP, Sander J, Xu X et al (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Kdd, vol 96, pp 226–231
8. Etzioni O, Cafarella M, Downey D, Kok S, Popescu AM, Shaked T, Soderland S, Weld DS, Yates A (2004) Web-scale information extraction in knowitall: (preliminary results). In: Proceedings of the 13th WWW. ACM, pp 100–110
9. Faruqui M, Dyer C (2014) Improving vector space word representations using multilingual correlation. In: EACL, pp 462–471
10. Feldman R, Fresco M, Goldenberg J, Netzer O, Ungar L (2007) Extracting product comparisons from discussion boards. In: Data mining, 2007. ICDM 2007. IEEE, pp 469–474
11. Frey BJ, Dueck D (2007) Clustering by passing messages between data points. Science 315(5814):972–976
12. Gurobi Optimization I (2016) Gurobi optimizer reference manual. http://www.gurobi.com. Accessed 24 Aug 2018
13. Hamilton WL, Leskovec J, Jurafsky D (2016) Diachronic word embeddings reveal statistical laws of semantic change. arXiv preprint arXiv:160509096
14. Hua M, Pei J, Fu AW, Lin X, Leung HF (2007) Efficiently answering top-k typicality queries on large databases. In: Proceedings of VLDB, VLDB endowment, pp 890–901
15. Huang X, Wan X, Xiao J (2012) Learning to find comparable entities on the web. Web Inf Syst Eng WISE 2012:16–29
16. Jain A, Pantel P (2009) Identifying comparable entities on the web. In: Proceedings of the 18th ACM CIKM. ACM, pp 1661–1664
17. Jiang Z, Ji L, Zhang J, Yan J, Guo P, Liu N (2013) Learning open-domain comparable entity graphs from user search queries. In: Proceedings of the 22nd ACM CIKM. ACM, pp 2339–2344
18. Jindal N, Liu B (2006) Identifying comparative sentences in text documents. In: Proceedings of ACM SIGIR. ACM, pp 244–251
19. Jindal N, Liu B (2006) Mining comparative sentences and relations. In: AAAI, vol 22, pp 1331–1336
20. Kaluarachchi AC, Varde AS, Bedathur S, Weikum G, Peng J, Feldman A (2010) Incorporating terminology evolution for query translation in text retrieval with association rules. In: CIKM. ACM, pp 1789–1792
21. Kanhabua N, Nørvåg K (2010) Exploiting time-based synonyms in searching document archives. In: JCDL. ACM, pp 79–88
22. Li S, Lin CY, Song YI, Li Z (2013) Comparable entity mining from comparative questions. IEEE TKDE 25(7):1498–1509
23. Lieberman E, Michel JB, Jackson J, Tang T, Nowak MA (2007) Quantifying the evolutionary dynamics of language. Nature 449(7163):713
24. Liu J, Wagner E, Birnbaum L (2007) Compare&contrast: using the web to discover comparable cases for news stories. In: Proceedings of the 16th WWW. ACM, pp 541–550
25. Lu A, Wang W, Bansal M, Gimpel K, Livescu K (2015) Deep multilingual correlation for improved word embeddings. In: NAACL HLT, pp 250–256
26. McCallum A, Jensen D (2003) A note on the unification of information extraction and data mining using conditional-probability, relational models
27. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv preprint arXiv:13013781
28. Mikolov T, Le QV, Sutskever I (2013) Exploiting similarities among languages for machine translation. arXiv preprint arXiv:13094168

29. Pagel M, Atkinson QD, Meade A (2007) Frequency of word-use predicts rates of lexical evolution throughout Indo-European history. Nature 449(7163):717
30. Pennington J, Socher R, Manning C (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
31. Reed SE, Zhang Y, Zhang Y, Lee H (2015) Deep visual analogy-making. In: Advances in neural information processing systems, pp 1252–1260
32. Řehůřek R, Sojka P (2010) Software framework for topic modelling with large corpora. In: Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks, ELRA, Valletta, Malta, pp 45–50. http://is.muni.cz/publication/884893/en. Accessed 10 Oct 2018
33. Rodríguez MA, Egenhofer MJ (2003) Determining semantic similarity among entity classes from different ontologies. IEEE TKDE 15(2):442–456
34. Sandhaus E (2008) The New York times annotated corpus overview. The New York Times Company, Research and Development, pp 1–22
35. Sarawagi S, Cohen WW (2005) Semi-Markov conditional random fields for information extraction. In: NIPS, pp 1185–1192
36. Scott DW, Sain SR (2005) Multidimensional density estimation. Handb. Stat 24:229–261
37. Smith SL, Turban DH, Hamblin S, Hammerla NY (2017) Offline bilingual word vectors, orthogonal transformations and the inverted softmax. arXiv preprint arXiv:170203859
38. Tahmasebi N, Gossen G, Kanhabua N, Holzmann H, Risse T (2012) Neer: an unsupervised method for named entity evolution recognition. In: COLING, pp 2553–2568
39. Tamma V, Bench-Capon T (2002) An ontology model to facilitate knowledge-sharing in multi-agent systems. Knowl Eng Rev 17(1):41–60
40. Wan X, Yang J (2008) Multi-document summarization using cluster-based link analysis. In: Proceedings of ACM SIGIR. ACM, pp 299–306
41. Wang Q, Mao Z, Wang B, Guo L (2017) Knowledge graph embedding: a survey of approaches and applications. IEEE Trans Knowl Data Eng 29(12):2724–2743
42. Xing C, Wang D, Liu C, Lin Y (2015) Normalized word embedding and orthogonal transform for bilingual word translation. In: NAACL HLT, pp 1006–1011
43. Yu HT, Jatowt A, Blanco R, Joho H, Jose J, Chen L, Yuan F (2017) A concise integer linear programming formulation for implicit search result diversification. In: Proceedings of the tenth ACM WSDM. ACM, pp 191–200
44. Zhang Y, Jatowt A, Bhowmick S, Tanaka K (2015) Omnia mutantur, nihil interit: Connecting past with present by finding corresponding terms across time. In: ACL vol 1, pp 645–655