Typicality-based Across-time Mapping of Entity Sets in Document Archives

Yijun Duan[1(✉)], Adam Jatowt[1], Sourav S. Bhowmick[2], and Masatoshi Yoshikawa[1]

[1]Graduate School of Informatics, Kyoto University, Japan
[2]Nanyang Technological University, Singapore
{yijun, adam}@dl.kuis.kyoto-u.ac.jp
yoshikawa@i.kyoto-u.ac.jp
assourav@ntu.edu.sg

Abstract. News archives constitute a rich source of knowledge about the past societies. In order to effectively utilize such large and diverse accounts of the past, novel approaches need to be proposed. One of them is comparison of the past and present entities which can lay grounds for better comprehending the past and the present, as well as can support forecasting techniques. In this paper, we propose a novel research task of *automatically generating across-time comparable entity pairs given two sets of entities*, as well as we introduce an effective method to solve this task. The proposed model first applies the idea of typicality analysis to measure the representativeness of each entity. Then, it learns an orthogonal transformation between temporally distant entity collections. Finally, it generates a set of typical comparables based on a concise integer linear programming framework. We experimentally demonstrate the effectiveness of our method on the New York Times corpora through both qualitative and quantitative tests.

Keywords: Comparable entity mining; temporal embeddings alignment; integer linear programming

1 Introduction

Comparison is an effective strategy extensively adopted in practice to discover commonalities and differences between two or more objects. Users can benefit from comparison for a myriad of needs such as understanding complex concepts, gaining insights about similar objects/situations, making better decisions and so on. Entity comparison has been studied in the past (e.g., [31]). 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].

Sometimes, however, what users want is to compare different entity sets across time. For example, a journalist or historian may be interested in the comparison of contemporary politicians with ones of 30 years ago. Another example could be a comparison of electronic gadgets used in 1980s-1990s with

those used at present, i.e., 2000s-2010s, that a student or scholar in the history of science/technology might wish to conduct. This kind of temporal analogy determination could be beneficial for general understanding of the relation of and the similarity between the past and present. Furthermore, it could then lead to not only improved comprehension of the past but could also complement and support our forecasting abilities. Manual comparison of entity sets across time is however non-trivial due to the following reasons: (1) In the current fast-paced world, people tend to possess limited knowledge about things from the past. In other words, it is difficult for average users to find temporal comparable entities (e.g. to know that music device like *Walkman* was playing a similar role 30 years ago as *iPod* does nowadays.); (2) such comparison involves entire entity sets, which is not an easy task given their diversity and complexity, thus it would require much cognitive effort. Note that entity sets are quite common in the real world and can be massive. For example, Wikipedia, which is considered to be the most comprehensive encyclopedia, contains over 1.13 million categories grouping numerous entities and concepts in many diverse ways [1].

A natural method of comparison is to find pairs of corresponding entities (e.g., finding a set of representative pairs of similar politicians or corresponding technological devices from among temporally distant time periods). Indeed, learning from examples is regarded an effective strategy extensively adopted in daily life. Good examples are often easier to be understood for learning concepts or categories of entities than high-level feature descriptions. Therefore, given two collections of entities from different historical times (e.g. the lists of contemporary politicians and the one of politicians active 30 years ago), it would be useful to automatically find a diverse set of corresponding entity pairs (e.g., U.S. Presidents: *Donald Trump* and *Ronald Reagan*, Russian Presidents: *Vladimir Putin* and *Mikhail Gorbachev*) as such pairs do not only provide contrasting information, but can be also understandable and intuitive.

Users can benefit from our study with respect to many needs. First of all, our study paves the way for the automatic discovery of mapping relationships between exemplars, which gives rise to the entity analogy solving task. Solving analogy tasks and generating analogical examples can be then enhanced using our method. Besides, finding typical comparables is a natural prerequisite step of discovering the commonalities and differences.

The problem of automatically detecting comparable entity pairs is however non-trivial due to the following reasons: (1) To measure across-time entity correspondence is a difficult task. The general context of the two compared entity collections which originate from different time periods may be fairly different. Intuitively, the correspondence of entities in different contexts cannot be computed properly without a solid understanding of the connection (analogies) between their contexts. Moreover, it is difficult to collect training data for learning such connections. (2) Naturally, only typical entities should be chosen for comparison. This is because typical instances are usually associated with more representative features and thus are less likely to cause misunderstanding. For instance, to compare mammals with another animal class, typical examples of mammals

such as lions should be preferred rather than atypical instances like platypuses (which lay eggs instead of giving birth). (3) The input sets of entities can be very diverse and may cover multiple latent subgroups. Thus instead of a single output entity pair, a set of pairs that represent latent subgroups within the input entity sets should be returned. Given the limitation on the size of output, selecting a subset of optimal pairs is a challenging problem since they should contain both typical and temporally comparable entities.

In view of the above-mentioned challenges, we propose a novel method to address the task of generating *typical comparables*. First of all, we formulate the measurement of *entity typicality* inspired by research in psychology and cognitive science [38, 6, 14]. In particular, for an entity to be typical in a diverse set it should be representative within a significant subset of that set. Moreoever, we formulate the measurement of across-time entity comparability by aligning different vector spaces and finding corresponding terms. We first adopt the distributed vector representation [27] to represent the context vectors of entities; then we learn linear and orthogonal transformations between two vector spaces of input collections for establishing across-time entity correspondence. Finally, inspired by the popular Affinity Propagation algorithm (AP) [11], we propose a concise joint integer linear programming framework (J-ILP) which detects typical entities (which we call exemplars) and, at the same time, generates comparable pairs from the detected exemplars. Based on this formulation, the optimal solution can be obtained.

To sum up, we make the following contributions: (1) We introduce a new research problem of automatically discovering comparable entity pairs from two across-time collections of entities. (2) We develop a novel method to address this task based on an efficient entity typicality estimation, an effective across-time entity comparability measurement, and a concise integer linear programming framework. (3) Finally, we perform extensive experiments on the New York Times Annotated Corpus, which demonstrates the effectiveness of our approach.

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 $P = [p_1, p_2, ..., p_m]$ 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.

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 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

$Typ(e, S)$. The entities to be selected for comparison should be typical in their sets, namely, $Typ(e_i^A, D_A)$ and $Typ(e_i^B, D_B)$ 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 $Typ(e, S)$ considering the characteristics of our task. First of all, the collections of entities for comparison can be very complex, 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 $[S^1, S^2, ..., S^k]$, 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:

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

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 non-parametric density estimations. We use kernel estimation [3] as it does not require any distribution assumption and can estimate unknown data distributions effectively. Moreover, we choose the commonly used Gaussian kernels. We set the bandwidth of the Gaussian kernel estimator $h = \frac{1.06s}{\sqrt[5]{n}}$ as suggested in [35], where n is the size of the data and s is the standard deviation of the data set. Formally, given a set of entities $S = (e_1, e_2, ..., e_n)$, 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*.

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 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). Thus our task is then to build the connection between semantic spaces of D_A and D_B.

Let transformation matrix W map the entities from D_A into D_B, and transformation matrix Q map the entities in D_B back into D_A. Let a and b be normalized entity representations from D_A and D_B, respectively. The comparability between entities a and b can be evaluated as the similarity between vectors b and Wa, i.e., $Comp(a, b) = b^T Wa$. However we could also form this correspondence as $Comp'(a, b) = a^T Qb$. To be self-consistent, we require $Comp(a, b) = Comp'(a, b)$, thus the linear transformations W and Q between entity collections D_A and D_B should be orthogonal [40, 36], i.e., $W^T W = I$ (where I denotes the identity matrix).

Our task is then to train the transformation matrix W to automatically align the semantic vector space across time. We adopt here a technique proposed by [42] for preparing sufficient training data. Namely, we use so-called Common Frequent Terms (CFT) as the training term pairs. CFT are very frequent terms in both dates T_A and T_B, which the compared entity collections originate from (e.g. man, woman, sky, water). 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 [23, 29, 13]. We first train the time-aware distributional vectors of CFTs using the New York Times Corpus [32] published within T_A and T_B, respectively. Given L pairs composed of normalized vectors of CFTs trained in both news corpora [(a_1,b_1), (a_2,b_2), ..., (a_L,b_L)] (where a_i and b_i denote the vector of i-th CFT in T_A and T_B, respectively), we should learn the transformation W by maximizing the accumulated cosine similarity of CFT pairs,

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

The solution corresponds to the best rotational alignment [34] and can be obtained efficiently using an application of SVD. By computing the SVD of $M = A^T B = U \Sigma V^T$, the optimized transformation matrix W^* satisfies $W^* = U \cdot V^T$. 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:

$$Comp(e_A, e_B) = Sim_{cosine}(W^* \cdot e_A, e_B) \tag{4}$$

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 are m comparable entity pairs [p_1, p_2, ..., 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. (5)) suffers considerably from convergence issues [41]. 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.

Specifically, we 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 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, $M_T = [m_{ij}]^T$ 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. 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:

$$\begin{aligned} max \;\; & \lambda \cdot m \cdot [T'(M_A) + T'(M_B)] \\ & + (1-\lambda) \cdot (k_A + k_B) \cdot C'(M_T) \end{aligned} \tag{5}$$

$$T'(M_X) = \sum_{i=1}^{n_X} m_{ii}^X \cdot Typ(e_i^X, G(e_i^X)), X \in \{A, B\} \tag{6}$$

$$C'(M_T) = \sum_{i=1}^{n_A} \sum_{j=1}^{n_B} m_{ij}^T \cdot Comp(e_i^A, e_j^B) \tag{7}$$

$$G(e_i^X) = \left\{ e_j^X | m_{ji}^X = 1, j \in \{1, ..., n_X\} \right\}, \\ i \in \{1, ..., n_X\}, X \in \{A, B\} \tag{8}$$

$$s.t. \quad m_{ij}^X \in \{0, 1\}, i \in \{1, ..., n_X\}, \\ j \in \{1, ..., n_X\}, X \in \{A, B\} \tag{9}$$

$$m_{ij}^T \in \{0, 1\}, i \in \{1, ..., n_A\}, j \in \{1, ..., n_B\} \tag{10}$$

$$\sum_{i=1}^{n_X} m_{ii}^X = k_X, X \in \{A, B\} \tag{11}$$

$$\sum_{j=1}^{n_X} m_{ij}^X = 1, i \in \{1, ..., n_X\}, X \in \{A, B\} \tag{12}$$

$$m_{jj}^X - m_{ij}^X \geq 0, i \in \{1, ..., n_X\}, \\ j \in \{1, ..., n_X\}, X \in \{A, B\} \tag{13}$$

$$\sum_{i=1}^{n_A} \sum_{j=1}^{n_B} m_{ij}^T = m \tag{14}$$

$$m_{ii}^A - m_{ij}^T \geq 0, i \in \{1, ..., n_A\}, j \in \{1, ..., n_B\} \tag{15}$$

$$m_{jj}^B - m_{ij}^T \geq 0, i \in \{1, ..., n_A\}, j \in \{1, ..., n_B\} \tag{16}$$

$$\sum_{j=1}^{n_B} m_{ij}^T \leq 1, i \in \{1, ..., n_A\} \tag{17}$$

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

We now explain the meaning of the above formulas. First, Eq. (11) forces that k_A and k_B exemplars are identified for both sets D_A and D_B, respectively, and Eq. (14) guarantees that m entity pairs are selected as the final result. The restriction given by Eq. (12) means each entity must choose only one exemplar. Eq. (13) 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 (15) and (16) jointly guarantees that if an entity is selected in any comparable entity pair (i.e., $m_{ij}^T = 1$), then it must be an exemplar in its own subgroup (i.e., $m_{ii}^A = 1$ and $m_{jj}^B = 1$). Restricted by Eq. (17) and Eq. (18), 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 [1] 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 [32]. 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 [4]. For the experiments, we first 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 then set on comparing the pair of time periods which are separated by the longest time gap, [1987, 1991] (denoted as T_A) and [2002, 2007] (denoted as T_B). 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 obtain the distributed vector representations for time period T_A and ones for T_B by training the Skip-gram model using the gensim Python library [30]. The number of dimensions of word vectors is experimentally set to be 200.

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 [2] for recognizing named entities based on all unigrams and bigrams. In total we extract 68,872 entities and 34,151 entities in T_A ([1987, 1991]) and T_B ([2002, 2007]), respectively. The details of identified entities are shown in Tab. 1. The meaning of sub-categories can be found at spaCy website[3]. Note that some sub-categories of entities were not used due to their weak significance, e.g., TIME/DATE.

Table 1. Summary of datasets.

heightPeriod	LOC	PRODUCT	NORP	WOA	GPE	PERSON	FACT	ORG	LAW	EVENT	TOTAL
T_A	427	87	2,959	129	7,810	33,127	328	23,775	18	212	68,872
T_B	304	44	1,573	91	4,460	16,103	221	11,215	11	129	34,151

[1] We experimentally set the value of λ to be 0.4 in Sec "Experiments".
[2] https://github.com/explosion/spaCy
[3] https://spacy.io/api/annotation#named-entities.

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 Sec. 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, 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 or use search engines in order to verify the correctness of the results. In total, 447 entities and 315 entities were chosen as typical exemplars for periods T_A and T_B, respectively. Among them, 168 pairs were constructed.

6.3 Evaluation Criteria

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.
Criteria for qualitative evaluation To further evaluate the quality of the results we also conducted user-based analysis. In particular, 3 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 [5] (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.

$$\mathrm{P} \equiv \ \mathrm{argmax} \sum_{i=1}^{m} Comp(e_i^A, e_i^B) \tag{19}$$

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 performing 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 [39] we set the number k of latent subgroups of each input set as:

$$k = \lceil \sqrt{n} \rceil \tag{20}$$

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 upper bound $min\{k_A, k_B\}$, where k_A and k_B are the numbers of identified exemplars of two compared entity sets.

(3) **number of used CFTs**: Following [42] we utilize the top 5% ($\approx 18k$) of *Common Frequent Terms* to train the orthogonal transformation in Sec. 3.

6.6 Evaluation Results

Results of quantitative evaluation Table 2 shows the performance of all the analyzed methods in terms of *Precision*, *Recall* and F_1*-score*, while we show the detailed results for a few examples in Tab. 3. 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 Tab. 2 reveals that regardless of the type of evaluation metric, J-ILP improves the performance of the other models under transformation. From Tab. 2, it can be seen that 27.3% entity pairs generated by J-ILP model are judged as correct by human annotators, and that 29.0% of ground truth entity pairs are discovered. Specifically, J-ILP improves the baselines by 87.3% when measured using the main metric F_1-*score* on average. 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.

Table 2. Performance of models in terms of *Precision*, *Recall* and F_1-*score*. 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		
	Precision	Recall	F_1-score	Precision	Recall	F_1-score	Precision	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**

Results of qualitative evaluation Fig 1 shows the evaluation scores in terms of *Correctness*, *Comprehensibility* and *Diversity* judged by annotators, 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*, respectively. This observation proves that J-ILP has relatively good performance in detecting 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 by 10.0% and 18.2%, respectively. It may be because AP algorithm and I-ILP are intrincically better in capturing representative and diverse exemplars, while J-ILP aims to balance the entity typicality and comparability simultaneously.

Table 3. Example results where entity pairs are ground truth. 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.

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)

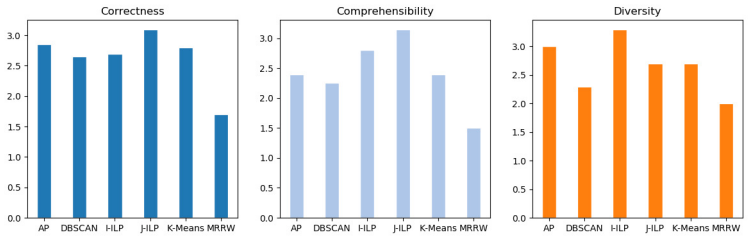

Fig. 1. Qualitative evaluation of results.

6.7 Additional Observations

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. 2, 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 this study we set λ as 0.4 based on the observations received from Fig. 2.

Sensitivity to kernel choice In this work we adopt Gaussian kernel function for computing entity typicality. Let the generated pairs returned by using Gaus-

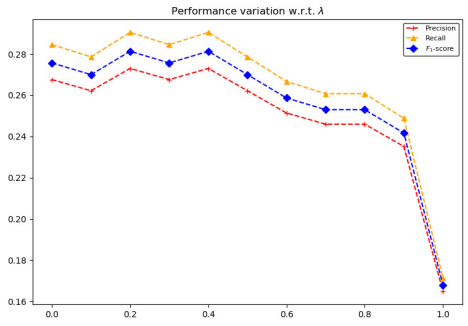

Fig. 2. Performance variation of *precision*, *recall* and F_1*-score* w.r.t. λ.

sian 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.

$$d = \frac{|P_G - PO|}{|PG|} \cdot 100\% \tag{21}$$

Table 4. Difference rate vs. kernel function.

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

Table 4 shows that the exemplars identified by different kernels are in general consistent, as the difference rate d is low.

7 Related Work

Comparable Entity Mining. The task of comparable entity mining has attracted much attention in the NLP and Web mining communities [15, 17, 22, 18, 19, 16]. Approaches to this task include hand-crafted extraction rules [8], supervised machine-learning methods [26, 33] and weakly-supervised methods [22, 17]. Jindal *et al.* [18, 19] was 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 [24, 10].

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 include computing term similarity across time [21, 37, 2, 20]. 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 bi-lingual word vectors for automatic text translation such as translation from Spanish to English [28]. Faruqui *et al.* obtained bi-lingual word vectors using CCA [9]. More recently, Xing *et al.* argued that the linear matrix adopted by Mikolov *et al.* should be orthogonal [40]. Similar suggestion has been given by Samuel *et al.* [36]. Besides linear models, non-linear models such as "deep CCA" has also been introduced for the task of mapping multi-lingual 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 Conclusions and Future Work

Entity is an evolving construct. This fact is nicely portrayed by the Latin proverb: *omnia mutantur, nihil interit* (in English: everything changes, nothing perishes) which indicates that there are no completely static things [42]. Across-time comparison based on typical 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). In this work, we propose a novel research problem of automatically detecting across-time typical comparable entity pairs from two input sets of entities and we introduce effective method for solving it. 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 plan to test our model on more heterogeneous datasets where contexts of entities are more difficult to be compared. We will also modify our model for query-sensitive comparative summarization tasks benefiting from high flexibility of the proposed ILP framework.

9 Acknowledgements

This research has been supported by JSPS KAKENHI grants (#17H01828, #18K19841). We thank the anonymous reviewers for their insightful comments.

References

1. Bairi, R.B., Carman, M., Ramakrishnan, G.: On the evolution of wikipedia: Dynamics of categories and articles. In: AAAI (2015)
2. Berberich, K., Bedathur, S.J., Sozio, M., Weikum, G.: Bridging the terminology gap in web archive search. In: WebDB (2009)
3. Breiman, L., Meisel, W., Purcell, E.: Variable kernel estimates of multivariate densities. Technometrics **19**(2), 135–144 (1977)
4. Campos, R., Dias, G., Jorge, A.M., Jatowt, A.: Survey of temporal information retrieval and related applications. ACM Computing Surveys (CSUR) **47**(2), 15 (2015)
5. Chen, Y.N., Metze, F.: Two-layer mutually reinforced random walk for improved multi-party meeting summarization. In: SLT, 2012 IEEE. pp. 461–466. IEEE (2012)
6. Dubois, D., Prade, H., Rossazza, J.P.: Vagueness, typicality, and uncertainty in class hierarchies. International Journal of Intelligent Systems **6**(2), 167–183 (1991)
7. Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Kdd. vol. 96, pp. 226–231 (1996)
8. Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S., Yates, A.: Web-scale information extraction in knowitall:(preliminary results). In: Proceedings of the 13th WWW. pp. 100–110. ACM (2004)
9. Faruqui, M., Dyer, C.: Improving vector space word representations using multilingual correlation. In: EACL. pp. 462–471 (2014)
10. Feldman, R., Fresco, M., Goldenberg, J., Netzer, O., Ungar, L.: Extracting product comparisons from discussion boards. In: Data Mining, 2007. ICDM 2007. pp. 469–474. IEEE (2007)
11. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. science **315**(5814), 972–976 (2007)
12. Gurobi Optimization, I.: Gurobi optimizer reference manual (2016), http://www.gurobi.com
13. Hamilton, W.L., Leskovec, J., Jurafsky, D.: Diachronic word embeddings reveal statistical laws of semantic change. arXiv preprint arXiv:1605.09096 (2016)
14. Hua, M., Pei, J., Fu, A.W., Lin, X., Leung, H.F.: Efficiently answering top-k typicality queries on large databases. In: Proceedings of VLDB. pp. 890–901. VLDB Endowment (2007)
15. Huang, X., Wan, X., Xiao, J.: Learning to find comparable entities on the web. Web Information Systems Engineering-WISE 2012 pp. 16–29 (2012)
16. Jain, A., Pantel, P.: Identifying comparable entities on the web. In: Proceedings of the 18th ACM CIKM. pp. 1661–1664. ACM (2009)
17. Jiang, Z., Ji, L., Zhang, J., Yan, J., Guo, P., Liu, N.: Learning open-domain comparable entity graphs from user search queries. In: Proceedings of the 22nd ACM CIKM. pp. 2339–2344. ACM (2013)
18. Jindal, N., Liu, B.: Identifying comparative sentences in text documents. In: Proceedings of ACM SIGIR. pp. 244–251. ACM (2006)
19. Jindal, N., Liu, B.: Mining comparative sentences and relations. In: AAAI. vol. 22, pp. 1331–1336 (2006)
20. Kaluarachchi, A.C., Varde, A.S., Bedathur, S., Weikum, G., Peng, J., Feldman, A.: Incorporating terminology evolution for query translation in text retrieval with association rules. In: CIKM. pp. 1789–1792. ACM (2010)

21. Kanhabua, N., Nørvåg, K.: Exploiting time-based synonyms in searching document archives. In: JCDL. pp. 79–88. ACM (2010)
22. Li, S., Lin, C.Y., Song, Y.I., Li, Z.: Comparable entity mining from comparative questions. IEEE TKDE **25**(7), 1498–1509 (2013)
23. Lieberman, E., Michel, J.B., Jackson, J., Tang, T., Nowak, M.A.: Quantifying the evolutionary dynamics of language. Nature **449**(7163), 713 (2007)
24. Liu, J., Wagner, E., Birnbaum, L.: Compare&contrast: using the web to discover comparable cases for news stories. In: Proceedings of the 16th WWW. pp. 541–550. ACM (2007)
25. Lu, A., Wang, W., Bansal, M., Gimpel, K., Livescu, K.: Deep multilingual correlation for improved word embeddings. In: NAACL HLT. pp. 250–256 (2015)
26. McCallum, A., Jensen, D.: A note on the unification of information extraction and data mining using conditional-probability, relational models (2003)
27. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
28. Mikolov, T., Le, Q.V., Sutskever, I.: Exploiting similarities among languages for machine translation. arXiv preprint arXiv:1309.4168 (2013)
29. Pagel, M., Atkinson, Q.D., Meade, A.: Frequency of word-use predicts rates of lexical evolution throughout indo-european history. Nature **449**(7163), 717 (2007)
30. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. pp. 45–50. ELRA, Valletta, Malta (May 2010), http://is.muni.cz/publication/884893/en
31. Rodríguez, M.A., Egenhofer, M.J.: Determining semantic similarity among entity classes from different ontologies. IEEE TKDE **15**(2), 442–456 (2003)
32. Sandhaus, E.: The new york times annotated corpus overview. pp. 1–22. The New York Times Company, Research and Development (2008)
33. Sarawagi, S., Cohen, W.W.: Semi-markov conditional random fields for information extraction. In: NIPS. pp. 1185–1192 (2005)
34. Schönemann, P.H.: A generalized solution of the orthogonal procrustes problem. Psychometrika **31**(1), 1–10 (1966)
35. Scott, D.W., Sain, S.R.: 9-multidimensional density estimation. Handbook of statistics **24**, 229–261 (2005)
36. Smith, S.L., Turban, D.H., Hamblin, S., Hammerla, N.Y.: Offline bilingual word vectors, orthogonal transformations and the inverted softmax. arXiv preprint arXiv:1702.03859 (2017)
37. Tahmasebi, N., Gossen, G., Kanhabua, N., Holzmann, H., Risse, T.: Neer: An unsupervised method for named entity evolution recognition. COLING pp. 2553–2568 (2012)
38. Tamma, V., Bench-Capon, T.: An ontology model to facilitate knowledge-sharing in multi-agent systems. The Knowledge Engineering Review **17**(1), 41–60 (2002)
39. Wan, X., Yang, J.: Multi-document summarization using cluster-based link analysis. In: Proceedings of ACM SIGIR. pp. 299–306. ACM (2008)
40. Xing, C., Wang, D., Liu, C., Lin, Y.: Normalized word embedding and orthogonal transform for bilingual word translation. In: NAACL HLT. pp. 1006–1011 (2015)
41. Yu, H.T., Jatowt, A., Blanco, R., Joho, H., Jose, J., Chen, L., Yuan, F.: A concise integer linear programming formulation for implicit search result diversification. In: Proceedings of the Tenth ACM WSDM. pp. 191–200. ACM (2017)
42. Zhang, Y., Jatowt, A., Bhowmick, S., Tanaka, K.: Omnia mutantur, nihil interit: Connecting past with present by finding corresponding terms across time. In: ACL. vol. 1, pp. 645–655 (2015)