Reconciling Event-Based Knowledge through RDF2VEC

Mehwish Alam¹, Diego Reforgiato Recupero²,³, Misael Mongiovì², Aldo Gangemi¹,², Petar Ristoski⁴


Abstract. The reconciled knowledge graphs are typically used for multidocument summarization, or to detect knowledge evolution across document series. This paper focuses on reconciling knowledge graphs generated from two text documents about similar events described differently. Our approach employs and extends MERGILO, a tool for reconciling knowledge graphs extracted from text, using word similarity and graph alignment. Complete semantic representation of events are generated using FRED, a semantic web machine reader, jointly with Framester, a linguistic linked data hub represented using a novel formal semantics for frames. Event-reconciliation is mainly performed via similarities based on the graph structure of frames using RDF2Vec graph embeddings, and the subsumption hierarchy of semantic roles as defined in Framester. Our approach is evaluated over a coreference resolution task.

Keywords: Knowledge Reconciliation, Event Reconciliation, Frame Embeddings, Frame Similarity, Role Similarity, Role Embeddings, Framester.

1 Introduction

This study targets the problem of knowledge reconciliation (KR) [18] from the perspective of events. KR is useful in providing a combination of multiple graphs generated by multiple texts describing the same event. This merged graph provides a graph based summary of multiple texts which is more easily comprehensible by users and machines and usable by the algorithms providing interactive exploration of graphs/text analytics through visualization methods.

MERGILO [18] is a tool for reconciling knowledge graphs extracted from text, it first computes the word similarity between the node labels and then performs graph alignment over the complete graphs. When different verbs denote similar events and different agents play slightly different roles, the string matching techniques as introduced in MERGILO might not be appropriate in the KR process. For overcoming this limitation we use Frame Semantics which describes a situation in the text with the help of frames and roles. For identifying frames and semantic roles of entities in a text we use FRED [12], a machine reader which generates event-centered knowledge graphs from two different texts.
Then, the similarity between these events is computed by calculating the similarity between the corresponding FrameNet \[2\] frames and semantic roles (frame elements). We adapt WordNet \[10\] similarity measures \[4\] to frames and roles and vector based similarities using the FrameNet graph and the subsumption hierarchy of roles as defined in Framester \[11\]. We follow the approach RDF2Vec \[23\] to generate graph based frame embeddings, used to calculate the semantic similarity between frames. It uses graph mining algorithms such as graph walks and graph kernels to traverse the graph for generating sequences, which are then fed to neural model for generating its vector representations. An evaluation on Cross-document coreference resolution shows significant improvement over the baseline.

The rest of this paper is structured as follows. Section 2 lists the data sources, resources and tools we have adopted in our methodology. Section 3 includes state of the art work. Then, Section 4 gives some details of MERGILO and its functionalities and explains how frame semantics have been employed for improving MERGILO. Section 5 shows a precision-recall analysis for the presented approach on the EECB dataset. Finally, Section 6 concludes the paper with discussions, remarks and highlights some future directions.

2 Role Oriented Resources

FrameNet \[2\] contains frames, which describe a situation, state or action. Each frame has frame elements usually consisting of agent, patient, time and location and are also known as semantic roles. Each frame can be evoked by Lexical Units (LUs) belonging to different parts of speech. These LUs can be nouns, verbs, adjectives and adverbs representing closely related sets of meanings. For example, in the frame Conquering the argument for the role Conqueror overtakes the argument of the role Theme where the theme loses its autonomy. Such constructs describing the situation of conquering or invasion are referred to as frame elements and the LUs such as conquer, overtake etc. are example words, typically used to denote conquering situations in text. In the example bellow, The Spaniards is the argument of the role Conqueror and Incas is the argument of the role Theme and conquered is the LU evoking the frame.

\[
\text{[The Spaniards]}\text{[Conqueror [conquered]}\text{Lexical Unit [the Incas]}\text{Theme]}.\]

Framester \[11\] is a large RDF\(^1\) knowledge graph (currently including about 30 million RDF triples) acting as a hub between FrameNet, WordNet, VerbNet \[14\], BabelNet \[19\], Predicate Matrix \[6\], etc. Framester uses a mapping between WordNet, BabelNet, VerbNet and FrameNet at its core using detour based approach, expands it to other linguistic resources transitively. It further links these

\(^1\) https://www.w3.org/TR/rdf11-primer/
resources to important ontological and linked data resources such as DBpedia, YAGO, DOLCE-Zero [20], schema.org etc.

Framester keeps the original FrameNet graph where the nodes represent the FrameNet frames and the edges represent different semantic relations between the frames i.e., Inheritance, SubFrame, CausativeOf etc. Figure 1 shows a part of FrameNet graph. Framester also contains a new subsumption hierarchy of semantic roles (i.e., frame elements) and added generic roles on top of the frame specific roles. Figure 2 shows a part of the Framester role hierarchy associated with the frame role agent.\footnote{The prefixes for http://www.ontologydesignpatterns.org/ont/framenet/abox/gfe/ and http://www.ontologydesignpatterns.org/ont/framester/data/framesterrole.ttl# are gfe: and framesterrole: respectively.}
FRED \cite{12} is a machine reader which generates ontological structure from natural language text using Discourse Representation Theory (DRT), frame semantics and Ontology Design Patterns. FRED uses Boxer,\footnote{http://wit.istc.cnr.it/stlab-tools/fred} an open source tool for deep parsing of natural language using Combinatory Categorial Grammar (CCG) and produces event-based, semantic representations of natural language. The Discourse Representation Structures (DRS) produced by Boxer use Verb-Net thematic roles. These functionalities implemented in FRED help in the event detection task for our method.

\section{3 State of the Art}

Approaches for integrating knowledge include cross-document coreference resolution (when knowledge is represented as text documents) and ontology matching (when knowledge is in a machine-readable form). Cross-document coreference resolution aims at associating mentions about a same entity (object, person, concept, etc.) across different texts \cite{8}. When extracted entities are events, the problem changes to resolution of event coreference across documents \cite{3}. The authors in \cite{16} jointly model named entities and events. Clusters of entities and event mentions are constructed and merged accordingly to a similarity threshold based on linear regression. Then, information flows between entity and event clusters through features that model semantic role dependencies. The system handles nominal and verbal events as well as entities, and the joint formulation allows information from event coreference to help entity coreference, and vice-versa. A rich overview of ontology matching methods is provided by \cite{9}. Relevant work includes \cite{24} that leverages the interplay between schema and
instance matching. Similarly, [15] shows a greedy iterative algorithm for aligning knowledge bases with millions of entities and facts. These approaches are characterised by the preferred large size of the ontologies/datasets treated (for best performance), which is rarely (probably never) derived from text sources. MERGILO, as other knowledge integration tools [15], employs graph alignment, a more general and widely studied problem [26]. Note that all these approaches are connected and related to the classical graph matching problem [22]. We address this problem from the perspective of events, by taking advantage of frame embeddings i.e., the vector representations of linguistic frames and semantic roles.

Recently, word embeddings have been used in variety of Information Retrieval and Natural Language Processing applications. One recent application is used for generating vector representations of word senses [13] and then these vector representations are used for improving the results of word similarity and word analogy tasks based on BabelNet word senses formally known as SensEmbed. [5] apply Frame Semantics and Distributional Semantics for slot filling in Spoken Dialogue System. In [27], the authors use Word and Frame Embeddings for generating categories of annoying behaviors where each category contains a set of words specific to that category. The frame embeddings are generated using 3.8 million tweets tagged by FrameNet frames using SEMAFOR. By contrast, in this study we are using graph-based Frame and Role Embeddings.

4 Event-Based Knowledge Reconciliation

Consider the two sentences: **Sent1**: “The Spaniards conquered the Incas.” and **Sent2**: “The Incas were attacked by the Spaniards.” They are describing the same event in the past using different words i.e., event of an attack or an invasion from Spaniards to Incas. Figure 3 shows the FRED graph of **Sent1**. Given two such knowledge graphs, MERGILO first performs graph compression by merging the nodes in the same graph. The two compressed graphs are aligned by establishing a 1-1 correspondence between the nodes of the two graphs by maximizing a score function, which combines the similarity between aligned nodes and the similarity between aligned edges. In such a case, the similarity between “conquered” and “attacked” is not effective since the word similarity is low, although in this context such words describe the same event.

For computing similarity between two nodes containing verb senses, the verb senses are first mapped to frames using Framester mappings. For example, in Figure 3 $s_1 \equiv \text{vn.data: Conquer}_42030000$ and for **Sent2** we have $s_2 \equiv \text{vn.data: Attack}_33000000$. According to Framester mappings, we obtain $s_1 \rightarrow \{\text{Conquering}\}$ and $s_2 \rightarrow \{\text{Attack}\}$. These nodes are replaced by their corresponding frames. The edges containing the VN-roles are mapped to FN-roles. For example, in Figure 3, the verb sense $\text{vn.data: Conquer}_42030000$ evokes the roles $\text{vn.role:Agent}$ and $\text{vn.role:Patient}$ which are mapped to $\text{fe:Conqueror.conquering}$ and $\text{fe:Theme.conquering}$ respectively.

5 prefix vn.data: http://www.ontologydesignpatterns.org/ont/vn/vn31/data/
In the sentence in Figure 2, the roles evoked by the verb sense vndata:Attack are vndata:Agent and vndata:Theme. The Framester mappings contains the following records for these roles:

- vndata:Theme.attack_33000000 skos:closeMatch fe:Victim.attack

Then the similarities are computed in two ways: (i) by considering the taxonomical structure imposed by the “inheritance” relation represented as fnschema:inheritsFrom in Framester using Path Similarity, Wu-Palmer Similarity, Leacock-Chodorow Similarity; (ii) using Frame Embeddings.

Frame Embeddings using RDF2Vec: To learn latent numerical representation of the frames and roles in the FrameNet graph, we follow the RDF2Vec approach. First we transform the graph into a set of sequences of entities, which is then fed into a neural language models, resulting into vector representation of all the nodes in the graph in a latent feature space.

To convert the graph into a set of sequences of entities we use two approaches, i.e., graph walks and Weisfeiler-Lehman Subtree RDF Graph Kernels. (i) Graph Walks: given a graph \( G = (V, E) \), for each vertex \( v \in V \), we generate all graph walks \( P_v \) of depth \( d \) rooted in vertex \( v \). To generate the walks, we use the breadth-first algorithm. In the first iteration, the algorithm generates paths by exploring the direct outgoing edges of the root node \( v_r \). In the second iteration, for each of the previously explored edges, the algorithm visits the connected vertices. The final set of sequences for the given graph \( G \) is the union of the sequences of all the vertices \( P_G = \bigcup_{v \in V} P_v \). (ii) Graph Kernels: it computes the number of sub-trees shared between two or more graphs by using the Weisfeiler-Lehman [7] test of graph isomorphism. This algorithm creates labels representing subtrees.

Once the set of sequences of entities is extracted, we build a word2vec model. Word2vec is a particularly computationally-efficient two-layer neural net model for learning word embeddings from raw text. There are two different algorithms, the Continuous Bag-of-Words model (CBOW) and the Skip-Gram model. The CBOW model predicts target words from context words within a given window, while the skip-gram model does the inverse. Once the training is finished, the cosine similarity is computed between two frames and roles.

5 Evaluation

The experiments were conducted for the task of Cross-document Coreference Resolution (CCR) on RDF graphs, which focuses on associating RDF nodes about a same entity (object, person, concept, etc.) across different RDF graphs generated from text. The data set used for the experimentation was obtained from:

6 prefix fnschema: http://www.ontologydesignpatterns.org/ont/framenet/tbox/
by the EECB data set which specifies coreferent mentions (text fragment). Our
dataset was obtained by generating RDF graphs using FRED and associating
text mentions to graph nodes by manual annotations. The framework is built
on top of the original MERGIGO code, which was released as a Python tool[7].
IBM ILOG CPLEX 12.6.1 was used for solving the Integer Linear Program and
the experiments were conducted on a MacOS server with 6-Core Intel Xeon E5
3.50GHz and 64GB of RAM. We used the following metrics for evaluation: (i)
MUC [25]: Link-based metric that quantifies the number of merges necessary to
cover predicted and gold clusters; (ii) B3 [1]: Mention-based metric that quan-
tifies the overlap between predicted and gold clusters for a given mention; (iii)
CEAFM (Constrained Entity Aligned F-measure Mention-based) [17]: Mention-
based metric based on a one-to-one alignment between gold and predicted
clusters; (iv) CEAFE (Constrained Entity Aligned F-measure Entity-Based) [17]:
Entity-based metric based on a one-to-one alignment between gold and predicted
clusters; (v) BLANC (Bilateral Assessment of NounPhrase Coreference) [21]:
Rand-index-based metric that considers both coreference and non-coreference
links.

For the current evaluation, MERGIGO was considered as a baseline. Table 1
shows the results for the baseline method, the Wu-Palmer’s similarity, the Path
similarity, the Leacock-Chodorow similarity and the results for cosine similarity
using (i) graph walks and (ii) graph kernels with FrameNet roles respectively.
Here Frame2Vec refers to the vector representations generated for FrameNet
frames and Role2Vec refers to the vector representations generated for frame
elements i.e., semantic roles.

For the first approach with graph walks, for each entity in the FrameNet
graph 200 and 500 random walks were generated, each of depth 4 and 8. For
each entity in the subsumption hierarchy of roles we generate 400 random walks
with depth 4. For the Weisfeiler-Lehman algorithm, we use $h = 2$ iterations and
subgraph depth $d = 2$, and after each iteration of the algorithm we extract all
walks for each entity with the same depth. We use these sequences to build both
CBOW and Skip-Gram models with the following parameters: window size = 5;
number of iterations = 10; negative sampling for optimization; negative samples
= 25; with average input vector for CBOW. We experiment with 200 and 500
dimensions for the entities’ vectors.

The results clearly indicate that each model used for graph walks and graph
kernels performs better than the MERGIGO baseline for all the considered met-
rics, showing a clear advantage of using the proposed frame similarities for recon-
ciling knowledge graphs. The Wu-Palmer, Path and Leacock Chodorow measures
use the inheritance relations only whereas Frame2Vec employs either graph walks
or graph kernels over the FrameNet frame graph as well as subsumption hierar-
chy of FrameNet roles using either only FrameNet roles or improved subsumption
hierarchy of FrameNet roles as introduced in Framester. Based on these settings,
vector representations are generated which are further used for computing the
cosine similarity. In general, Frame2Vec, for its intrinsic construction, exploits

<table>
<thead>
<tr>
<th></th>
<th>muc</th>
<th>bcub</th>
<th>ceafm</th>
<th>blanc</th>
<th>ceafe</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MERGILO Baseline</strong></td>
<td>24.05</td>
<td>17.36</td>
<td>28.61</td>
<td>10.70</td>
<td>26.20</td>
</tr>
<tr>
<td><strong>Similarity Measures</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wu-Palmer</td>
<td>27.14</td>
<td>19.91</td>
<td>31.91</td>
<td>12.81</td>
<td>29.41</td>
</tr>
<tr>
<td>Path</td>
<td>27.16</td>
<td>19.93</td>
<td>31.85</td>
<td>12.73</td>
<td>29.38</td>
</tr>
<tr>
<td>Leacock Chodorow</td>
<td>27.04</td>
<td>19.80</td>
<td>31.74</td>
<td>12.77</td>
<td>29.21</td>
</tr>
<tr>
<td><strong>Graph walks</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frame2Vec</td>
<td>27.34</td>
<td>19.99</td>
<td>32.15</td>
<td>12.66</td>
<td>29.82</td>
</tr>
<tr>
<td>Role2Vec</td>
<td>27.38</td>
<td>19.97</td>
<td>32.29</td>
<td>12.69</td>
<td>29.98</td>
</tr>
<tr>
<td>CBOW_200</td>
<td>27.28</td>
<td>19.95</td>
<td>31.99</td>
<td>12.69</td>
<td>29.54</td>
</tr>
<tr>
<td>CBOW_200 SG_500</td>
<td>27.09</td>
<td>19.03</td>
<td>29.95</td>
<td>11.91</td>
<td>28.97</td>
</tr>
<tr>
<td>CBOW_200 CBOW_500</td>
<td>26.90</td>
<td>19.68</td>
<td>31.58</td>
<td>12.60</td>
<td>29.08</td>
</tr>
<tr>
<td>SG_200</td>
<td>26.87</td>
<td>19.57</td>
<td>31.33</td>
<td>12.10</td>
<td>29.01</td>
</tr>
<tr>
<td>SG_500</td>
<td>26.85</td>
<td>19.45</td>
<td>31.12</td>
<td>12.08</td>
<td>28.98</td>
</tr>
<tr>
<td><strong>Graph kernels</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frame2Vec</td>
<td>26.76</td>
<td>19.57</td>
<td>31.50</td>
<td>12.45</td>
<td>29.06</td>
</tr>
<tr>
<td>Role2Vec</td>
<td>26.76</td>
<td>19.57</td>
<td>31.50</td>
<td>12.45</td>
<td>29.06</td>
</tr>
<tr>
<td>CBOW_500</td>
<td>26.76</td>
<td>19.51</td>
<td>31.45</td>
<td>12.45</td>
<td>28.96</td>
</tr>
<tr>
<td>SG_200</td>
<td>26.86</td>
<td>19.62</td>
<td>31.67</td>
<td>12.48</td>
<td>29.18</td>
</tr>
<tr>
<td>SG_500</td>
<td>26.90</td>
<td>19.68</td>
<td>31.58</td>
<td>12.60</td>
<td>29.08</td>
</tr>
</tbody>
</table>

Table 1: Event-Based Knowledge Reconciliation Results. The best results are marked in bold.

more semantics than the other similarity measures (Wu-Palmer, Path and Leacock Chodorow); for such a reason, Frame2Vec provides the highest results for almost each evaluation measure except for BLANC.

BLANC is more sensitive to wrong assignments when clusters of mentions are larger, since a wrong assignment lead to a higher number of wrong non-coreference links. Therefore, although BLANC is case-by-case coherent with the other measures (when BLANC is low, the other measures are low and vice-versa), in the few cases when Frame2Vec is outperformed by other measures (Wu-Palmer, Path and Leacock Chodorow), the BLANC measure, and in particular the contribution given by non-coreference link, gives a much smaller score. These cases influence the overall average and for this reason in Table 1 BLANC seems to have a different behaviour than the other measures.

The generated models i.e., vector representations of FrameNet frames generated using FrameNet graph and subsumption hierarchy of FrameNet roles using RDF2Vec are freely available on-line.\(^8\)

6 Conclusions and Discussion

This paper presents a way to perform event-reconciliation for merging multiple event-oriented knowledge graphs originated from multiple texts. It uses existing tool MERGILO, a tool for reconciling knowledge graphs using word similarity and graph alignment. The current study exploits several path-based similarity measures for frames and semantic roles, i.e., following the approach RDF2Vec, graph-based frame embeddings were generated. The evaluation shows that the introduced approach is an effective improvement over the baseline.

Ongoing work concentrates on practical applications of frame embeddings in real systems, such as news series integration, knowledge graph evolution with robust event reconciliation (e.g. in streaming of texts where we expect relatedness or updates), or conflict detection across texts describing similar facts with different narratives or perspectives.

References