Jing Zhang

Jing Zhang[1], Bonggun Shin[2], Jinho D. Choi[1], and Joyce C Ho[1]

[1] Emory University, Atlanta GA 30329, USA
{jing.zhang2, jinho.choi, joyce.c.ho}@emory.edu
[2] Deargen Inc., Seoul, South Korea
bonggun.shin@deargen.me

Abstract. Schema matching aims to identify the correspondences among attributes of database schemas. It is frequently considered as the most challenging and decisive stage existing in many contemporary web semantics and database systems. Low-quality algorithmic matchers fail to provide improvement while manually annotation consumes extensive human efforts. Further complications arise from data privacy in certain domains such as healthcare, where only schema-level matching should be used to prevent data leakage. For this problem, we propose SMAT, a new deep learning model based on state-of-the-art natural language processing techniques to obtain semantic mappings between source and target schemas using only the attribute name and description. SMAT avoids directly encoding domain knowledge about the source and target systems, which allows it to be more easily deployed across different sites. We also introduce a new benchmark dataset, OMAP, based on real-world schema-level mappings from the healthcare domain. Our extensive evaluation of various benchmark datasets demonstrates the potential of SMAT to help automate schema-level matching tasks.

Keywords: Schema-level matching · Natural language processing · Attention over attention.

1 Introduction

The tremendous growth and availability of data can benefit a broad range of applications such as healthcare, energy, transportation, and smart buildings. Unfortunately, across many domains, data is collected using a wide variety of database systems with customized schemas developed for each company or purpose. The customized databases can hinder data exchange, data integration, and large-scale analytics. Schema matching aims to establish the correspondence between the fields of a source and target database schema – a decisive initial step in the standardization of different databases. Automation of the schema matching task has received steady attention in the database and AI communities over the years. It has also been adopted as a practical and principled tool to improve the modeling and implementation of data exchange and data integration [2,22,27]. Yet, this problem remains largely unsolved and still requires significant manual labor.

Given the importance of schema matching and the time-intensive nature of the task, it is crucial to develop new methods to help expedite the process. Several automated schema matching methods have been proposed, including constraint-based approaches [5,13,34] and linguistic-based approaches [18,21,24,39]. While the existing methods have achieved high performance in different domains, they suffer from several limitations. The constraint-based approaches analyze the element contents, which is not always guaranteed to be the same across the two schemas. Moreover, it assumes the data on both sides can be queried, which can violate privacy constraints. For the linguistic approaches, the relations are hand-coded between the two schemas or may not properly capture the similarity between the field descriptions. Numerous matching tools (matchers) can generate correspondences between pairs of schemas [6,13]. Yet they rely on heuristic techniques. Recently, a deep neural network (DNN)-based model, ADnEV, was proposed to utilize similarity from existing matchers and post-process the results to work across domains [36]. However, ADnEV is limited by the capability of existing matchers and may not generalize to all domains.

Given the rising importance of schema integration involving sensitive data, such as in healthcare, we focus on schema-level matching rather than instance-level or hybrid schema matching. This paper posits that the schema matching process (i.e., source schema elements to target schema elements and its attributes matching) can be viewed as inferring the relatedness (or similarity) between the source and target fields. We propose `SMAT`, a DNN-based model with attention that extends recent advances in natural language processing and sentiment analysis. `SMAT` captures the semantic correlation from the source schema attributes to the target schema attributes based on the name and descriptions. Moreover, our model can be used to automatically generate the matching between the source and target schemas without encoding domain knowledge. We also introduce a new publicly available dataset that annotates several source to target conversions in the healthcare domain. We perform extensive evaluations of SMAT on a variety of datasets.

2 Related Work

This section describes the existing works related to schema-level matching that only considers schema information and not instance data. For a detailed survey on schema matching, we refer the reader to [34]. Table 1 provides a brief comparison of some related works and our model along four categories (i.e., whether it is schema-level matching, what the match cardinality is, whether it captures rich text, and whether it utilizes deep-learning framework).

One line of schema matching work is the constraint-based approach. Most schemas contain constraints to define the attributes such as data types and value ranges, uniqueness, optionality, relationship types and cardinalities [34]. Similarity can be measured by data types and domains, key characteristics (e.g., unique, primary, foreign), and relationship cardinality [1,14,29]. Recently, [3] proposes a hybrid of the constraint-based approach using key characteristics and

Table 1: Comparison between different approaches on various categories.

Approach	Schema-level	Cardinality	Rich text	Deep learning
Constraint-based [3]	No	1:n	No	No
Linguistic content-based [24]	Yes	n:1	No	No
ADnEV [36]	Yes	n:1	No	Yes
DITTO [27]	No	n:1	Yes	Yes
SMAT	Yes	n:1	Yes	Yes

the instance itself to create the meta-schema. Unfortunately, such approaches cannot readily handle the n:1 scenario that can be found in schema matching. For example, if the source schema contains "starttime" and "endtime" and the target schema contains "Duration", the meta-schema mapping can not generate and convert the two attributes into the single target.

An alternative method is the linguistic content-based approach, which utilizes names and text to explore semantically similar schema elements. There are two primary linguistic data mapping techniques: name matching and description matching. The idea behind these techniques is to calculate similarity based on either the name of the fields or the description of the fields, respectively. In name matching, the similarity of names can be defined and measured through equality of names, equality of synonyms, similarity of names based on common substrings and user-provided name matches. Examples include [20] which helps database designers visualize similarity and dissimilarity based on attribute names and [40] which uses a prescribed dictionary to obtain the aggregation among attributes. However, consulting a synonym lexicon has limitations since it is common to use abbreviations for attribute names (e.g., DOB for date of birth, SSN for Social Security number, etc.) and may not identify the relationships.

Description matching is based on the idea that schemas usually contain element and attribute names in natural language to express the intended semantics of schema elements. The process involves the identification of two semi-related data objects and the creation of mappings between them. In a recent work [24], the authors utilized the UMBC EBIQUITY-CORE technique [19] to obtain the similarity of the comments of schemas. Yet, it may not capture the similarity between the descriptions. For example, the similarity score between "the comment of the book" and "the review of the article" is 0.39 (1 is the same and 0 is dissimilar). Another work used word embeddings to link datasets [15]; however it only embeds the table name which may not yield sufficient information.

With the development of DL techniques, entity matching [4,27], ontology alignment [25], and instance-level schema-matching [26] can utilize rich textual information to provide better solutions. However, both entity matching and instance-level schema matching assume the data can be queried on both sides, which can violate data privacy constraints. For schema-level matching, [31] proposed a probabilistic graphical model and achieved a good score on precision and recall. Recently, ADnEV was proposed to utilize a DL technique to post-process

the matching results from other matchers and outperformed existing models. However, the quality of the matchers limits the potential of the model.

3 SMAT: A DNN Model

We introduce SMAT, an attention-based DNN model to automate the schema matching between the source and target schemas. We posit that the attribute-to-attribute matching performed in schema matching can be viewed as inferring the relatedness (or similarity) between the source and target fields. Under this paradigm, the data dictionaries containing the tables and attributes descriptions can be used to automatically capture the semantic correlation between the two fields without requiring explicit domain knowledge. SMAT extends recent advances in natural language processing (NLP) and sentiment analysis to encode the field descriptions for both source and target and determine which two fields are related to one another. In this section, we formulate the problem and then introduce the various components of SMAT.

3.1 Problem Statement

Given two table descriptions S_{TS} and S_{TT}, two attributes names N_{F1} and N_{F2}, and their descriptions S_{F1} and S_{F2} from the source and target schema respectively, we construct two sets of sentences. The source sentence set $S_S = \{N_{F1}, S_{TS} + S_{F1}\} = \{w_1, w_2, ..., w_n\}$ consists of n words, and the target sentence set $S_T = \{N_{F2}, S_{TT} + S_{F2}\} = \{w_1, w_2, ..., w_{n'}\}$ consists of n' words. For the training data, there is an annotated label $L(S_S, S_T)$ where 0 denotes two fields are not related (i.e., not mapped to each other), and 1 denotes two sentences are related (i.e., corresponding attribute-to-attribute matching). Table 2 provides an example of the sentence pair. Thus the task objective is to classify the semantic relation of each sentence pair to reveal the attribute-to-attribute matching.

3.2 Overview

The task of determining the relatedness between two attributes descriptions can be viewed as inferring the similarity of two sentence pairs in NLP tasks. Since DNNs can be trained end-to-end without any prior knowledge (i.e., no need to implement feature engineering), they have been utilized for text similarity tasks. For sentiment classification, InferSent introduced an end-to-end DNN and achieved a higher performance than existing sentiment analysis models [8]. Yet there are two major limitations to adopting such models for the schema matching task. First, the element and attribute description may not contain sufficient information to distinguish it from others. Second, the descriptions may have abbreviations or words that have unknown word representations.

To address the above limitations, SMAT consists of 4 major modules (shown in Figure 1). First, the input embedding of the sentences utilizes a hybrid encoding

to deal with large vocabularies for any input text. Second, bidirectional Long short term memory (BiLSTM) networks are used to capture the hidden semantics of the words in the description and the column name separately. Third, the attention over attention (AOA) mechanism [9] is adopted to model the correlation between the column name and its description to obtain a better sentence representation.

The final prediction layer uses the sentence representations to make an accurate classification. We also introduce data augmentation and controlled batch sample ratios (CBSR) to deal with the class imbalance problem that is present in schema matching tasks.

3.3 Input Embedding & BiLSTM

Existing word embedding models such as GloVe [33] are limited by vocabulary size or the frequency of word occurrences. As a result, rare words like *ICD-9* result in unknown tokens. Byte-Pair Encoding (BPE) is a hybrid between character- and word-level representations which can deal with the large vocabularies common in natural language corpora [35]. Instead of full words, BPE learns sub-words units to tokenize any input text without introducing any "unknown" tokens.

Thus, `SMAT` uses BPE to tokenize the input text. Each word/sub-word w_i in the sentence $S_1 = \{w_1, w_2, ..., w_n\}$ is then mapped to a high-dimensional vector e_i, using GloVe embeddings. While we use GloVe due to its popularity, any word embedding representation can be used.

To capture the contextual nature of the text, a BiLSTM network is utilized to capture the hidden semantics. Compared with the standard LSTM, BiLSTM can utilize both the past and the future information to yield better sentence representations. Thus, after the word embedding is obtained for each set of words (i.e., attribute name or attribute description), the embeddings are fed to a BiLSTM network.

3.4 Attention-over-Attention (AOA)

The output of the BiLSTM is dealt with using two approaches. All the information in the sequence is captured using the max-pooling operator to compress the sequence into a single unified vector. However, one limitation of this representation is the inability to capture interactions between the attribute name and its description. The second approach uses an attention over attention (AOA) module to model this interaction. AOA was first proposed for the question answering task [9]. Since calculating the dot product and difference of two sentence representations fail to capture fine-grained relations on the word level, the AOA module introduces mutual attention to simultaneously capture the relationships between attribute name to description and description to attribute name.

Our AOA module captures the correlations between the attribute names and the text using two mechanisms. Let $h_c \in R^{m \times 2h}$ denote the attribute name representation, where m is the attribute name length (i.e., number of words in

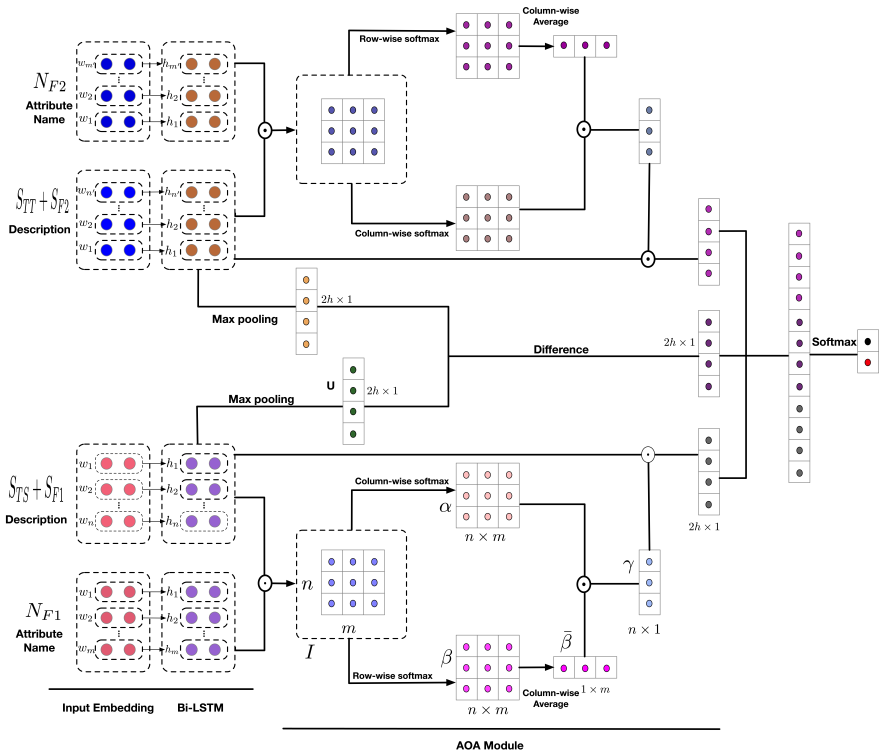


Fig. 1: Illustration of SMAT's structure

the attribute name) and h is the hidden dimension. Let $h_s \in R^{n \times 2h}$ denote the element-attribute description representation, where n is the description length and h is the hidden dimension. The module first calculates the pair-wise interaction matrix $I = h_s \cdot h_c^T$, where the value of each entry represents the correlation of each word pair between the description and attribute name. A column-wise softmax and row-wise softmax is applied to the interaction matrix I, to obtain the attribute name to description attention, α, and description to attribute name attention, β, respectively. Thus for the t^{th} attribute word and k^{th} text description, the associated attentions are:

$$\alpha(t) = softmax(I(1,t), I(2,t), \cdots, I(m,t)) \tag{1}$$

$$\beta(k) = softmax(I(k,1), I(k,2), \cdots, I(k,n)) \tag{2}$$

Then, the attribute name-level attention $\bar{\beta}$ is calculated using a column-wise averaging of β. This attention indicates the important words in the attribute name. Finally, the sentence-level attention $\gamma \in R^n$ can be obtained by a weighted sum of each individual attribute name to description attention α. By considering the contribution of each word explicitly, the AOA module learns the important

weights for each word in the sentence.

$$\alpha_{ij} = \frac{\exp(I_{ij})}{\sum_i \exp(I_{ij})}$$
$$\beta_{ij} = \frac{\exp(I_{ij})}{\sum_j \exp(I_{ij})}$$
$$\bar{\beta} = \frac{1}{n} \sum_i \beta_{ij}$$
$$\gamma = \alpha \cdot \bar{\beta}^T$$

The two sets of final description level attentions for the source and target, γ_s and γ_t, are concatenated along with the difference between the two max-pooled attribute description representations. The new vector representation, P, is sent to the final classification layer which consists of several fully-connected layers and a softmax layer to predict whether or not two sentences are related.

3.5 Data Augmentation & Controlled Batch Sample Ratio

As attribute-to-attribute mapping generally results in a skewed distribution, `SMAT` uses data augmentation and controlled batch sample ratio (CBSR) to achieve better predictive performance. Data augmentation occurs on two levels. First is to generate new positive samples using synonyms for different words in the descriptors. For example, an augmented sample may replace the word "uniquely" with "unambiguously" and "identify" with "describe". However, since the number of synonyms is limited, we utilize a second technique to improve the attribute name description. We use the part-of-speech (POS) tags for the descriptions and concatenate the identified nouns to enlarge the dataset safely.

Since `SMAT` uses batch SGD to learn the parameters, a batch can contain no positive samples and thus only properly learn the representation for negative samples. Thus, we controlled the ratio of positive samples in each batch size to ensure that our model learns from a few positive examples for each batch [12]. Note that since the positive samples are small, they are likely to be chosen repeatedly, while there is diversity in the negative samples.

4 OMAP: A New Benchmark Dataset

Since existing matching datasets only spans purchase orders, web forms, and bibliographic references, we created `OMAP`, a new benchmark schema-level matching dataset that annotates several source-to-target mappings in the healthcare domain. Healthcare data is collected worldwide using a wide variety of coding systems. To draw conclusions with statistical power and avoid systematic biases, a large number of samples should be analyzed across disparate data sources and patient populations. Such broad analyses requires data harmonization to a common data standard (e.g., the Observational Medical Outcomes Partnership

Table 2: An example entry from the OMAP dataset.

CDM schema	Source schema	CDM description (Des 1)	Source description (Des 2)	Label
person-person_id	beneficiary summary-desynpuf_id	the person domain contains records that uniquely identify each patient in the source data who is time at-risk to have clinical observations recorded within the source systems.a unique identifier for each person.	beneficiarysummary pertains to a synthetic medicare beneficiary. beneficiary code	1

(OMOP) Common Data Model (CDM) standard) to facilitate evidence gathering and informed decision making [32]. Since patient data cannot be queried due to privacy concerns, schema-level matching is of great importance. OMAP maps between three different healthcare databases and the OMOP CDM standard.

1. MIMIC-III [23]: A publicly available intensive care unit (ICU) relational database from the Beth Israel Deaconess Medical Center.
2. Synthea [38]: An open-source dataset that captures the medical history of over one million Massachusetts synthetic patients.
3. CMS DE-SynPUF [7]: A set of realistic claims data generated from 5% of Medicare beneficiaries in 2008.

For each dataset, the element table name with its descriptions and attribute column name with its descriptions are used to construct a sentence. The label is based on the final ETL design. If the table-column in the source schema was mapped to a table-column in the OMOP CDM the label is 1, otherwise it is 0. Table 2 provides one example from the OMAP dataset.

Table 3: Summary statistics of each conversion captured in OMAP.

Data source	# elements	# attributes	# positive labels	# sentence pairs
MIMIC	25	240	129	64080
Synthea	12	111	105	29637
CMS	5	96	196	25632

OMAP currently contains 121,689 matching pairs from three different datasets and is available publicly on Github[3]. The summary statistics for each of the three conversions are captured in Table 3.

Note that the dataset does not contain any patient information, only attributes and their descriptions.

Table 4: Summary statistics of the additional benchmark datasets used.

Data source	# elements	# related	# pairs	# Domains
Purchase Order[11]	50-400	659	63933	1
OAEI[4]	80-100	9494	825021	1
Web-forms[16]	10-30	5548	201769	18

5 Experiments

We designed the experiments to answer three key questions: (1) How *accurate* is SMAT in automating the schema matching? (2) How sensitive is SMAT to the training size? (3) How important are the different components of SMAT?

5.1 Dataset

We use the OMAP dataset to evaluate our proposed model (see Table 3 and Section 4). We also used three popular schema matching benchmark datasets as shown in Table 4.

Reference matches in these datasets were manually constructed by domain experts and considered as ground truth for our purposes. Experiments are performed per dataset consistent with existing schema matching papers [17,31,37]. For each dataset, 80% was used to train the initial prediction model, the 10% used to further tune the weights, and the remaining 10% used to evaluate the experiments.

5.2 Baseline Models

SMAT is evaluated against five baseline models. For data sensitivity purposes, we focused only on schema-level matching. The entity matching solutions that involve semantic relatedness technique are chosen to represent the existing schema matching or entity matching work.

- **ADnEV** [36]. A schema matching model that utilizes DNN to post-process results from state-of-the-art (SOTA) matchers in an iterative manner.
- **InferSent** [8]. A SOTA sentence embedding model that classifies the sentiment between two sentences. The last layer is modified to tackle a binary classification task. GloVe embeddings [33] are used for the input sentences.
- **DeepMatcher** [30]. An entity matching solution that customizes the Recurrent Neural Network (RNN) architecture to aggregate the attribute values then compares the aggregated representations of attribute values.

[3] https://github.com/JZCS2018/SMAT

[4] The OAEI competitions can be found at http://oaei.ontologymatching.org/2011/benchmarks/

Table 5: Comparison of precision (P), recall (R), and F1 (F) on the datasets.

Dataset	ADnEV			InferSent			DeepMatcher			DITTO			BERT			SMAT		
	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
MIMIC	0.08	34	0.16	9.8	76.9	17.4	0.04	38.1	0.09	0.3	46.2	0.6	0.4	84.6	0.7	**11.5**	**84.6**	**20.2**
CMS	0.49	44	0.97	20.8	80.0	32.9	0.31	60.7	0.62	2.4	40	4.5	2.4	55.0	4.5	**33.9**	**95.0**	**50.0**
Synthea	0.14	21	0.28	19.2	90.9	31.7	0.06	48.8	0.13	0.7	63.6	1.3	0.9	**100**	1.8	**24.4**	90.9	**38.5**
Purchase Order	**80**	77	**78**	14.3	59.6	23.1	48.9	80.2	60.8	54.5	98.6	70.2	54.0	98.2	69.7	57.9	**99.5**	73.2
OAEI	78	76	76	84.5	99.9	91.5	56.1	62.9	59.3	80.5	99.9	89.2	78.3	99.8	87.8	**87.8**	**99.9**	**93.5**
Web-forms	**81**	69	72	68.4	**99.8**	81.2	48.2	74.5	58.5	68.8	95.5	80	63.5	96.3	76.5	79.1	99.3	**88.1**
Average	34.3	49.9	32.5	33.6	78.2	43.3	22.0	56.8	25.8	29.7	69.4	35.4	28.6	**88.8**	34.7	**45.7**	87.0	**56.3**

- **DITTO** [27]. A SOTA entity matching model that cast the problem as a sequence-pair classification and fine-tunes RoBERTa [28], a pre-trained Transformer-based language model.
- **BERT** [10]. Bidirectional Encoder Representations from Transformers (BERT) has achieved SOTA results in many natural language understanding tasks. We fine-tuned the pre-trained BERT-base-uncased model on our datasets.

5.3 Experimental Setup

We implemented SMAT and the baseline models in Python 3.6 using PyTorch. Our code is made publicly available on Github[5]. Performances were measured on the Google Cloud Platform with Intel Xeon E5 v3 CPU @ 2.30Ghz, and a Nvidia Tesla K80 with 12 GB Video Memory.

For experiments in this paper, the embedding dimension is 300. The number of hidden units of BiLSTM is 1024 for InferSent and 300 for SMAT. For the classification model, we apply a fully connected layer with one hidden layer of 512 hidden units. Stochastic gradient descent is chosen as the optimize algorithm with a batch size of 64. The learning rate and weight decay are 0.1 and 0.99 for InferSent and 0.001 and 0.99 for SMAT. For AdnEV, DeepMatcher, DITTO, and fine-tuning BERT model, Adam is chosen as the optimization algorithm with a learning rate of 0.001, 0.001, $3e-5$, $2e-5$, respectively, and the batch size as 64, 64, 64, and 32 respectively. These parameters were obtained from initial experiments on a subset of the training data as they provided the most robust performance across multiple runs.

6 Results

6.1 Predictive Performance

Evaluation of SMAT with existing baseline models. Table 5 summarizes the results of the six models tested on the six datasets. We observe that the precision and recall varies depending on the dataset suggesting differences in the semantic content of their attribute names and descriptions. The results demonstrate

[5] https://github.com/JZCS2018/SMAT

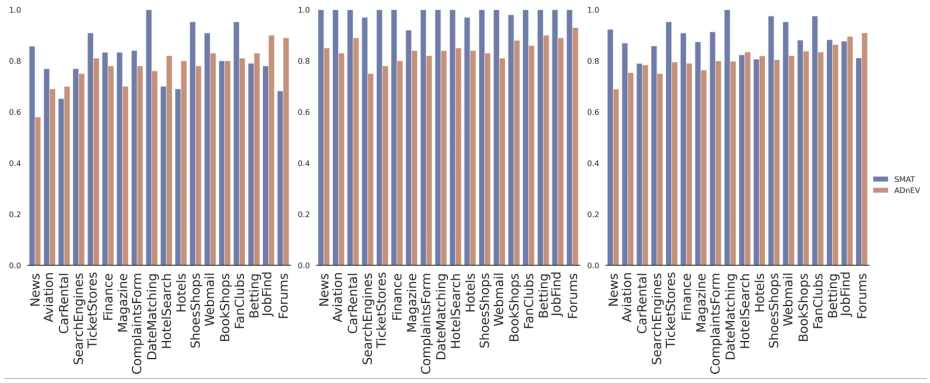
that SMAT does not require additional hand-coding due to the overall strong performance. It achieves the best performance across all three metrics in 3 of the datasets (OAEI, MIMIC, CMS). It also yields the best F1 score for all but the Purchase Order dataset. Thus, our proposed model is fairly versatile.

ADnEV achieves a higher precision on Purchase Orders and Webforms and a better F1 score on Purchase Orders than others. Yet, SMAT outperforms the ADnEV model on OAEI and Web-forms in terms of F1 score by 12.4% and 16.1% respectively. Moreover, the results on the OMAP datasets illustrate the pitfall of ADnEV. Since ADnEV leverages other matchers, it is limited by the capability of the matchers. Thus, ADnEV may not be suitable for all domains. Furthermore, comparisons of the DNN-based models (InferSent, Fine-tuned BERT, and SMAT) and ADnEV in terms of F1 and recall also illustrate the power of end-to-end training without requiring additional feature engineering.

For the OMAP dataset, SMAT achieves a higher precision and recall score suggesting that the prediction capability of SMAT is better than the other models. However the precision across these four datasets are noticeably lower than those of Purchase Order, OAEI and Web-forms. This may be a result of the more complex textual information in the healthcare domain. Moreover, there are many abbreviations which can prevent the general model from achieving a higher score. This highlights the importance of benchmarking the models across various applications and supports the development of OMAP.

The results also capture the difference that arises from schema-level matching. Even though DITTO and DeepMatcher perform well in the entity matching task, they do not offer comparable performance across the different datasets. This may be due to the inconsistencies across the datasets present in the textual information. Moreover, InferSent seems to provide better F1 scores compared to the more complex transformer models outside of the Purchase Dataset. This suggests that the Bi-LSTM based sentence modeling approach shared by InferSent and SMAT may offer better predictive power compared to the more complex transformer-based models. In comparing InferSent and SMAT, the results suggest that SMAT's attention mechanism and representation can help capture the elements and attributes in source schema and target schema differences better than the other models regardless of whether the textual information is rich (OMAP) or not (Purchase Order, OAEI and Web forms).

Analysis on Web-form, a cross-domain schema matching. The Webforms dataset contains 18 domains to represent the cross-domain matching task. Figure 2 analyzes the match quality per domain and compares the results between SMAT and ADnEV since ADnEV achieves the best precision. From the results, we observe that SMAT outperforms ADnEV across all the domains in terms of recall. Moreover, for the majority of the domains, SMAT offers better precision and F1 score over ADnEV. For example, the Webmail, Finance, and News domains are difficult for the ADnEV model. For example, existing matchers fail to identify the mappings *Measures the price performance of a stock in comparison to all other stocks (12 Month Relative Strength)* $\leftrightarrow$ *YTD total return* and *Mailbox* $\leftrightarrow$ *@gmail.com (Email)*. However, SMAT can capture the semantic



(a) *Precision* (P) (b) *Recall* (R) (c) $F1$ (F)

Fig. 2: Comparison by domain between ADnEV and SMAT

Table 6: Computational efficiency of the different methods on the Synthea dataset.

Model	Training time (sec/epoch)	Inference speed (sentence/second)
ADnEV	90	52
InferSent	86	275
DeepMatcher	113	209
DITTO	95	249
Fine-tuned BERT	127	186
SMAT	101	234

meaning of these pairs. The results also demonstrate that ADnEV performs better on the domain Forums and Hotels than SMAT. This is because SMAT excludes the number and type constraints in the element and attribute.

6.2 Computational Efficiency

Table 6 summarizes the training time and inference speed for the six different methods using the same computer hardware on Synthea dataset. Of all the DNN-based models, InferSent is the most computational efficient (i.e., lowest training time and highest inference speed). However, the quality of the prediction is significantly lower especially with respect to precision as shown in Table 5. With a slight increase in training and lower inference speed, SMAT provides the best overall predictive performance across the different datasets. It is also worthwhile to highlight that ADnEV takes less training time but the inference speed is substantially slower due to the need to post-process the results. Although DITTO is based on pre-trained Transformer-based language models, the

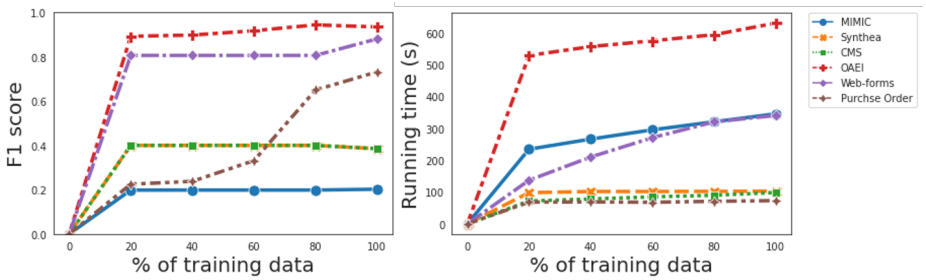


Fig. 3: F1 score (left) and running time (right) per epoch when varying (%) of training data

computation efficiency is better than fine-tuned BERT due to optimization details. Such optimization techniques can be adapted in our implementation for improved speed.

6.3 Training Size Sensitivity & Scalability

We assessed the robustness of `SMAT` to the size of the training data. We varied the amount of data used to fit `SMAT` and evaluate its impact on the test dataset performance. Figure 3 illustrates the results on the six different datasets in terms of F1 score and running time. From the results, we notice that `SMAT` achieves a decent F1 score with only about 20% training data (the lone exception is Purchase Orders) and can save 30% of the training time. We also notice that the running time per epoch is fairly linear suggesting that `SMAT` is scalable.

6.4 Ablation Study

To gain further insights of the various components in `SMAT`, we examined the effectiveness and contributions of the attribute name input, the AOA module, and the two different class imbalance approaches.

- *`SMAT` w/o AOA*: The AOA module is removed and instead the outputs of the attribute name BiLSTM and description BiLSTM are max-pooled together and concatenated with the difference of the two descriptions.
- *`SMAT` w/o column*: The attribute name is omitted and only the description is fed into the AOA module to calculate the mutual information with itself.
- *`SMAT` w/o DA*: The data augmentation with additional positive samples and concatenation of nouns to the column name is omitted.
- *`SMAT` w/o CBSR*: The batch size is randomly sampled without ensuring positive samples are present in each batch.

The results of the ablation study are shown in Table 7. It can be seen that the `SMAT` model outperforms the rest of four models on F1 and most precision. In particular, comparing the result with *`SMAT` w/o AOA* illustrates the importance of

Table 7: Results for ablation experiments on Precision (P), Recall (R), and F1 (F).

Dataset	SMAT			w/o AOA			w/o column			w/o DA			w/o CBSR		
	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
MIMIC	**11.5**	**84.6**	**20.2**	10.3	84.6	18.3	10.2	84.6	18.2	10.7	69.2	18.6	0	0	0
CMS	**33.9**	**95.0**	**50.0**	23.5	80.0	36.3	25.4	80.0	38.6	25.8	80.0	39.0	0.13	15.6	0.25
Synthea	24.4	90.9	**38.5**	15.3	90.0	26.1	20.0	**100**	33.3	**36.4**	36.4	36.4	0	0	0
Purchase Order	**57.9**	**99.5**	**73.2**	17.7	50.0	26.2	26.9	30.3	28.5	42.1	98.2	58.9	10.7	38.2	16.7
OAEI	**87.8**	**99.9**	**93.5**	83.0	99.9	90.7	83.8	99.9	91.2	85.9	99.9	92.4	35.9	72.5	48.0
Web-forms	**79.1**	**99.3**	**88.1**	75.7	96.7	84.9	76.4	93.5	84.1	70.0	99.8	82.3	32.5	68.4	44.1

the AOA module. The module captures the interaction between the attribute description and the correlated attribute name better than max-pooling the outputs from BiLSTM. The same conclusion can also be drawn by comparing *SMAT w/o AOA* and *SMAT w/o column*, the precision of the former is lower than the latter. Even without the attribute name feature and the associated data augmentation, the AOA module can still generate more useful features.

The ablation results also highlights two benefits of the model. First, the attribute name is important as there is a noticeable drop in precision across all the datasets when comparing *SMAT w/o column* with SMAT and *SMAT w/o D/A*. Second, the two techniques for dealing with class imbalance play a crucial role towards improving the predictive power of the model. The results of *SMAT w/o DA* and *SMAT w/o CBSR* shows that CBSR is more effective toward combating the skewed data than data augmentation method due to the higher precision values of the former model.

7 Conclusion

This paper proposes an automated schema-level matching model based on the semantic meaning of the descriptions. This is particularly beneficial for schema integration involving sensitive data, such as healthcare domain. The extensive experiments on a variety of datasets illustrate that SMAT serves as the SOTA solution for the schema-level matching task. This paper also introduces a new benchmark dataset, OMAP, that captures three different dataset conversions from the healthcare domain. As shown in the experiments, OMAP can help assess the generalizability of schema-level matching models.

Although the empirical results of SMAT are not yet high enough to be put into practice, this work illustrate the potential of automating schema matching. Future directions include collecting more data to improve the sentence embedding quality, exploring other DNN architectures to tackle the class imbalance problem, and incorporating instance-level features to obtain a robust hybrid schema-level and instance-level model.

Acknowledgements. This work was supported by the National Science Foundation award IIS-#1838200, National Institute of Health award 1K01LM012924, and Google Cloud Platform research credits.

References

1. Alexe, B., Hernández, M., Popa, L., Tan, W.C.: Mapmerge: Correlating independent schema mappings. Proceedings of the VLDB Endowment **3**(1-2), 81–92 (2010)
2. Arenas, M., Barceló, P., Libkin, L., Murlak, F.: Foundations of data exchange. Cambridge University Press (2014)
3. Atzeni, P., Bellomarini, L., Papotti, P., Torlone, R.: Meta-mappings for schema mapping reuse. Proc. VLDB Endow. **12**(5), 557–569 (Jan 2019). https://doi.org/10.14778/3303753.3303761
4. Cappuzzo, R., Papotti, P., Thirumuruganathan, S.: Creating embeddings of heterogeneous relational datasets for data integration tasks. In: Proc. of SIGMOD. pp. 1335–1349 (2020)
5. ten Cate, B., Kolaitis, P.G., Qian, K., Tan, W.C.: Active learning of GAV schema mappings. In: Proc. of SIGMOD/PODS. pp. 355–368 (2018)
6. Chen, C., Golshan, B., Halevy, A.Y., Tan, W.C., Doan, A.: Biggorilla: An open-source ecosystem for data preparation and integration. IEEE Data Eng. Bull. **41**(2), 10–22 (2018)
7. Centers for medicare & medicaid services (cms). `https://www.cms.gov/OpenPayments/Explore-the-Data/Data-Overview.html`
8. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A.: Supervised learning of universal sentence representations from natural language inference data. In: Proc. of EMNLP. pp. 670–680 (2017)
9. Cui, Y., Chen, Z., Wei, S., Wang, S., Liu, T., Hu, G.: Attention-over-attention neural networks for reading comprehension. In: Proc. of ACL (2017)
10. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proc. of NAACL-HLT. pp. 4171–4186 (2019)
11. Do, H.H., Rahm, E.: Coma—a system for flexible combination of schema matching approaches. In: Proc. of VLDB. pp. 610–621 (2002)
12. Dong, Q., Gong, S., Zhu, X.: Imbalanced deep learning by minority class incremental rectification. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(6), 1367–1381 (June 2019). https://doi.org/10.1109/TPAMI.2018.2832629
13. Fagin, R., Haas, L.M., Hernández, M., Miller, R.J., Popa, L., Velegrakis, Y.: Clio: Schema mapping creation and data exchange. In: Conceptual modeling: foundations and applications, pp. 198–236. Springer (2009)
14. Fagin, R., Kolaitis, P.G., Popa, L., Tan, W.C.: Schema mapping evolution through composition and inversion. In: Schema matching and mapping, pp. 191–222. Springer (2011)
15. Fernandez, R.C., Mansour, E., Qahtan, A.A., Elmagarmid, A., Ilyas, I., Madden, S., Ouzzani, M., Stonebraker, M., Tang, N.: Seeping semantics: Linking datasets using word embeddings for data discovery. In: Proc. of ICDE. pp. 989–1000 (2018)
16. Gal, A.: Uncertain schema matching. Synthesis Lectures on Data Management **3**(1), 1–97 (2011)
17. Gal, A., Roitman, H., Shraga, R.: Learning to rerank schema matches. IEEE Transactions on Knowledge and Data Engineering (2019)
18. Halevy, A., Nemes, E., Dong, X., Madhavan, J., Zhang, J.: Similarity search for web services. In: Proceedings of the 30th VLDB Conference. pp. 372–383 (2004)
19. Han, L., Kashyap, A.L., Finin, T., Mayfield, J., Weese, J.: Umbc_ebiquity-core: Semantic textual similarity systems. In: Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity. pp. 44–52 (2013)

20. Hayne, S., Ram, S.: Multi-user view integration system (muvis): An expert system for view integration. In: [1990] Proceedings. Sixth International Conference on Data Engineering. pp. 402–409. IEEE (1990)
21. He, B., Chang, K.C.C.: Statistical schema matching across web query interfaces. In: Proc. of SIGMOD. pp. 217–228 (2003)
22. Hernandez, M., Ho, H., Naumann, F., Popa, L.: Clio: A schema mapping tool for information integration. In: 8th International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN'05). pp. 1–pp. IEEE (2005)
23. Johnson, A.E., Pollard, T.J., Shen, L., Li-wei, H.L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., Mark, R.G.: Mimic-iii, a freely accessible critical care database. Scientific data **3**, 160035 (2016)
24. Kettouch, M.S., Luca, C., Hobbs, M., Dascalu, S.: Using semantic similarity for schema matching of semi-structured and linked data. In: 2017 Internet technologies and applications (ITA). pp. 128–133. IEEE (2017)
25. Kolyvakis, P., Kalousis, A., Kiritsis, D.: Deepalignment: Unsupervised ontology matching with refined word vectors. In: Proc. of NAACL-HLT. pp. 787–798 (2018)
26. Koutras, C., Fragkoulis, M., Katsifodimos, A., Lofi, C.: Rema: Graph embeddings-based relational schema matching. In: EDBT/ICDT Workshops (2020)
27. Li, Y., Li, J., Suhara, Y., Doan, A., Tan, W.C.: Deep entity matching with pre-trained language models. arXiv preprint arXiv:2004.00584 (2020)
28. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
29. Mecca, G., Papotti, P., Santoro, D.: Schema mappings: From data translation to data cleaning. In: A Comprehensive Guide Through the Italian Database Research Over the Last 25 Years, pp. 203–217. Springer (2018)
30. Mudgal Sunil Kumar, S.: Deep learning for entity matching: A design space exploration. Tech. rep. (2018)
31. Nguyen, Q.V.H., Weidlich, M., Nguyen, T.T., Miklós, Z., Aberer, K., Gal, A.: Reconciling matching networks of conceptual models. Tech. rep. (2019)
32. Observational Health Data Sciences and Informatics: The book of OHDSI. Independently published (2019)
33. Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. In: Proc. of EMNLP. pp. 1532–1543 (2014)
34. Rahm, E., Bernstein, P.A.: A survey of approaches to automatic schema matching. the VLDB Journal **10**(4), 334–350 (2001)
35. Sennrich, R., Haddow, B., Birch, A.: Neural machine translation of rare words with subword units. arXiv preprint arXiv:1508.07909 (2015)
36. Shraga, R., Gal, A., Roitman, H.: Adnev: cross-domain schema matching using deep similarity matrix adjustment and evaluation. Proc. of the VLDB **13**(9), 1401–1415 (2020)
37. Toan, N.T., Cong, P.T., Thang, D.C., Hung, N.Q.V., Stantic, B.: Bootstrapping uncertainty in schema covering. In: Australasian Database Conference. pp. 336–342. Springer (2018)
38. Walonoski, J., Kramer, M., Nichols, J., Quina, A., Moesel, C., Hall, D., Duffett, C., Dube, K., Gallagher, T., McLachlan, S.: Synthea: An approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record. Journal of the American Medical Informatics Association **25**(3), 230–238 (8 2017)

39. Wu, W., Yu, C., Doan, A., Meng, W.: An interactive clustering-based approach to integrating source query interfaces on the deep web. In: Proc. of SIGMOD. pp. 95–106 (2004)
40. Yu, C., Sun, W., Dao, S., Keirsey, D.: Determining relationships among attributes for interoperability of multi-database systems. In: [1991] Proceedings. First International Workshop on Interoperability in Multidatabase Systems. pp. 251–257. IEEE (1991)