International Journal of Artificial Intelligence and Machine Learning
Volume 11 • Issue 1 • January-June 2021

A Method Based on a New Word Embedding Approach for Process Model Matching

Mostefai Abdelkader, Dr. Tahar Moulay University of Saida, Algeria

Mekour Mansour, LabRI Laboratory, Ecole Superieure en Informatique, Sidi Bel Abbes, Algeria

ABSTRACT

This paper proposes a method based on a new word embedding approach for matching business process model. The proposed method aligns two process models in four steps. First activity labels are extracted and pre-processed to remove meaningless words, then each word composing an activity label and using a semantic similarity metric based on WordNet is represented with an n-dimensional vector in the space of the vocabulary of the two labels to be compared. Based on these representations, a vector representation of each activity label is computed by averaging the vectors representing words found in the activity label. Finally, the two activity labels are reported as similar if their similarity score computed using the cosine metric is greater than some predefined threshold. An experiment was conducted on well-known dataset to assess the performance of the proposed method. The results showed that the proposed method shared the first place with RMM/NHCM and OPBOT tools and can be effective in matching process models.

KEYWORDS

Alignment, Matching, Process Model, Vector Representation, Word Embedding, WordNet

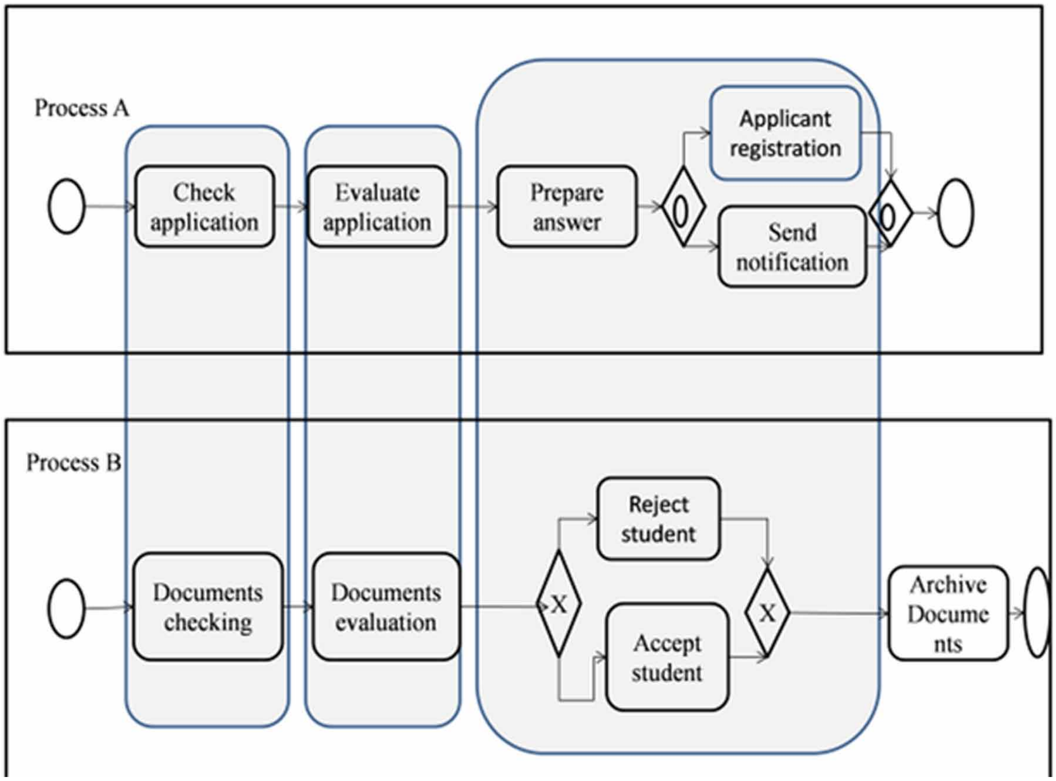
INTRODUCTION

Process model matching(i.e., PMM)is an activity widely conducted in organizations to align process models. This alignment is critical for many business process (i.e., BP) management tasks such as storing, merging, clustering or querying Business Process models (i.e., BPs) (La Rosa et al. 2013; Goncalo et al. 2015; Weidlich et al. 2012).

Technically an alignment is a set of correspondences between activities of two process models. Each correspondence is a pair of two semantically similar sets of activities. The first set of a pair contains activities from the first BP and the second set of this pair contains activities from the second BP. Formally, an alignment is a set of not overlapped matches pairs (i.e., correspondences) $\{(A_{11}, A_{21}),(A_{1.2},A_{22}),\ldots,(A_{1n},A_{2n})\}$. Each pair defines a match (i.e., correspondence) between a set of activities, A_{1i}, from BP_1 and a set of activities, A_{2i}, from BP_2. Two pairs (A_{1i}, $A_2 i$) and (A_{1j}, A_{2j}) does not overlap iff$A_{1i} \cap A_{1j} = \varnothing$ and $A_{2i} \cap A_{2j} = \varnothing$. A correspondence ($A_{1i}$, A_{2i}) between a set of activities A_{1i} from

DOI: 10.4018/IJAIML.2021010101

Figure 1. An example of an alignment



one BP and a set of activities A_{2i} means that the activities A_{1i} and A_{2i} refer to the same activity in the domain. Figure 1 presents of an example of an alignment between two process models.

Correspondences between the activities are presented using the grey shades. Examples of correspondence is between the set {*prepare answer, applicant registration, send notification*} in the process A and the set {reject *students, accept student}* in the business process B.

The objective of any process model matching technique is to find automatically such correspondences.

Many approaches have been proposed to achieve this objective(Goncalo et al. 2015; Daniel et al. 2017, Xue, 2019; Khurram et al. 2019). The proposed approaches are widely based on a combination of lexical, syntactic and semantic similarity metrics proposed in different fields such as the NLP (i.e., Natural language Processing) community (Goncalo et al. 2015; Daniel et al. 2017).

The literature showed that these approaches perform better in the context of ontology matching (Euzenat & Shvaiko, 2007) where they were proposed first and that they are less effective when adapted to the context of process model matching (Daniel et al., 2017). Thus matching process models stills an open problem and work needs to be done to find effective techniques for BP models matching (Goncalo et al. 2015; Cayoglu et al. 2013; Meilicke et al. 2017). The difficulty is caused by the vocabulary mismatch between activities in process models and the level of details of these activities. This means that similar activities in business process models are often described using different words and with different level of details (Weidlich et al. 210). For example, the activity "check order" could be represented in the other BP using different words as "verify request" and the activity "inform applicant" could be described using the following activities "send acceptance letter, send reject letter".

The difficulty arises also from the nature of activity label which tend to be short text and ambiguous (J. Mendling et al. 2009). For example, *check documents* or *send letters*.

It is important to note that the problem of detecting similar texts (short or long) is a long-researched problem in the NLP and information Retrieval domains (Eneko et al. 2016; Yates & Neto, 1999). Despite that many approaches have been proposed to solve this problem, it stills an open problem (Gomaa & Fahmy, 2013). These approaches are based on many word to word similarity measures that calculate a score of similarity between two words. These techniques are classified into three approaches corpus based grouped into two sub categories statistical analysis approaches and deep learning approaches, knowledge based (e.g., Wordnet based) or string based.

Many of these techniques was used in building process model matchers and their evaluation showed it that they are less effective and no one can be of practical use since no matchers was able to outperforms all the other in all cases (Goncalo et al. 2015). The best F-measure obtained was between 0.253-0.667 (Goncalo et al. 2015).

Thus, this paper proposes a method for matching process models. This method is based on a new word embedding approach. The proposed method is 4 steps and works as follow: first activity labels composing the two process models are extracted and pre-processed to remove meaningless words, then each word composing the activity label is represented, using a semantic similarity metric based on WordNet (Miller, 1995), with an N-dimensional vector in the space of the vocabulary of the two labels. The proposed word embedding approach is based on the assumption that two words are semantically similar if they share the same properties (e.g., semantic similarity) with the other words. Based on this representation each activity label is represented with N-dimensional vector by averaging its constituents' words vectors. Finally, the two activity labels as reported are similar if their similarity score computed using the cosine metric is greater than some predefined threshold.

An experiment was conducted on well-known datasets to assess the performance of the proposed method. The results showed that our method could be effective in matching process models.

The rest of the paper is organised as follows: section II presents the related work, section III presents the proposed approach, section IV presents the empirical study and discuss the obtained results, section V conclude and presents future works.

STATE OF THE ART

Many approaches have been proposed to find a good alignment. The proposed approaches are widely based on a combination of lexical, syntactic and semantic similarity metrics proposed in different fields such as the NLP (i.e., Natural language Processing) community (Goncalo et al. 2015; Daniel et al. 2017). Generally, these metrics are used to compute the similarity score between words composing activity labels and then these scores are combined using some strategy to decide if two activity labels are similar or not. Many of the proposed approaches was proposed as techniques to match ontologies and then they are evaluated on the context of process matching (Manel et al. 2017; Euzenat & Shvaiko 2007). The literature showed that these approaches perform better in the context of ontology matching and that they are less effective in the context of process model matching (Daniel et al. 2017).

The Process Model Matching Contests held in 2013 and 2015 highlighted the importance of process model matching (Goncalo et al. 2015; Cayoglu et al. 2013). The aims of the contests was the evaluation of the performance of the state of the art techniques proposed for PMM.

In (Khurram et al. 2019), the authors evaluated the effectiveness of many word embedding techniques proposed in the literature in matching process models. The authors found that fastText-based embeddings technique is effective than both Word2vec and GloVe-based embeddings techniques.

The TripleS (Cayuglu et al. 2013b; Goncalo et al. 2015) technique uses a combination of weighted syntactic, semantic and structural similarity metrics to compare two activity labels. The two labels are reported as similar if the computed score is greater than some predefined score.

Daniel et al (2013) presented The AgreementMakerLight (AML) system. AML combines the result of several matching algorithms. These algorithms exploit lexical and structural properties of process models.

In 2013 (Weidlich et al., 2013a), Weidich presented the BPLangMatch technique, which was based on a language modelling approach to detect matches where each process is viewed as a document and activity names as passages. The Jensen-Shannon divergence (JSD) similarity metric is used to calculate the similarity between passages in the documents (i.e., activities). The produced scores are modified according to the part of speech tagger information and The top-k strategies is used to derive the final correspondences (Avigdor & Tomer, 2010).

The RefMod-Mine/NLM matcher (Goncalo et al. 2015) uses natural language processing techniques and a bag of words concept model to create correspondences between BPs. Many matchers use this technique but the new in the RefMod-Mine/NLM approach is the use of a word classification approach.

The KnoMa-Proc system (Weidlich et al. 2013) uses information retrieval techniques to infer matches. The KnoMa-Proc technique starts by building an index containing a structured description of each entity. Second a retrieval technique finds best matches for each entity in the process model by considering each entity to be matched in the first BP as query on the index of entities in the second one and vice versa.

In (Goncalo et al. 2015; Abderrahmane khiat et al. 2015) presented the Match-SSS and Know-SSSS (MSSS) approach to align BPs. The system is based on string-based and WordNet similarity metrics to compare activities after a pre-processing step.

In Niesen et al (2015) the authors presented The RefMod-Mine/VM^2 technique to align business process models. It is a three steps technique, first a trivial match is done to find identical labels and labels that are substring of others. Then, a Lemma-based match is conducted on the harmonized labels and finally a Vector-based match is executed.

The RefMod-Mine/NHCM system is an amelioration of RefMod-Mine/NSCM presented at the Process Model contest 2013. It is three steps approach: First, models are transformed to a common format to allow the matching of different models written in different language. Second, a n-ary cluster matcher based on natural language-based similarity measure is used to compare pair of nodes and produces a binary complex mapping. Third a final alignment is produced based on the produced binary complex mappings and using several techniques.

In the pPalm-DS system (Goncalo et al. 2015) a vector representing each label is created by summing the vectors representing its composing words. The word vectors are computed using the word2Vec approach. Two labels are reported as similar if their similarity score computed using the cosine metric is greater than some predefined threshold.

The Order Preserving Bag-Of-Words Technique (OPBOT) (Goncalo et al. 2015) finds matches in two steps. First it detects activities having the same labels and then the details of the rest of the labels are reduced. This reduction is based on three similarity metrics. A syntactic one and is based on the longest common subsequence similarity. The paradigmatic similarity which based on Lin's similarity metric. The syntagmatic one that uses the cooccurrence counts.

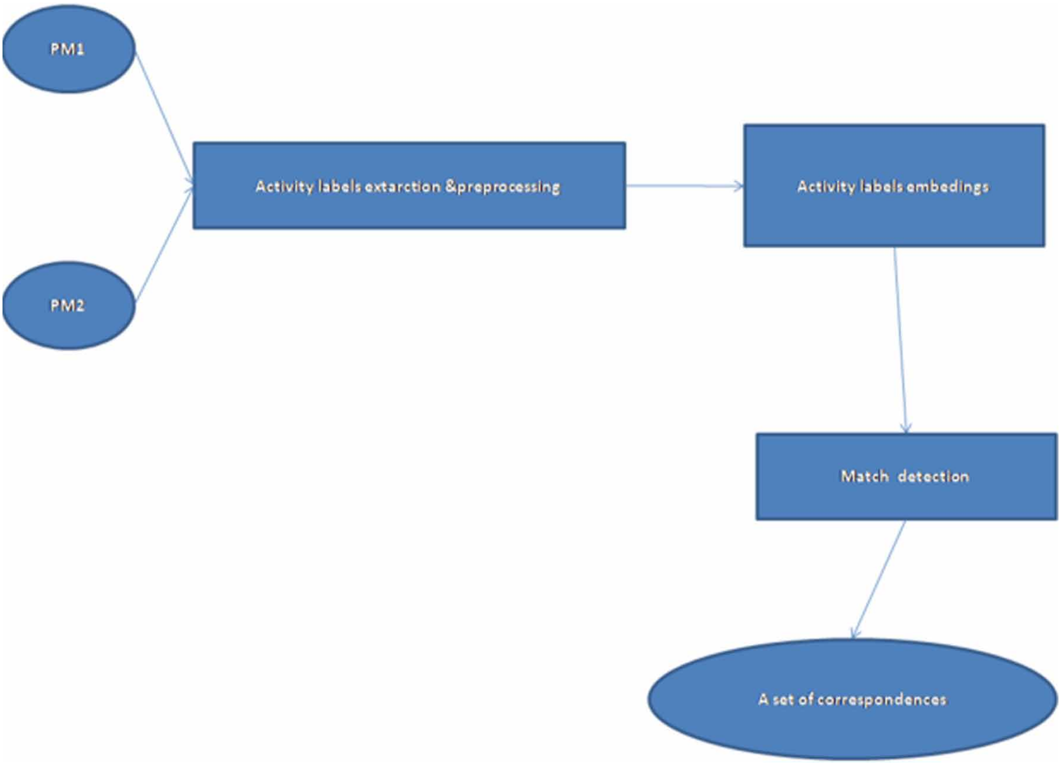
The RefMod-Mine/SMSL matcher (Goncalo et al. 2015) is a semantic matching algorithm based on a supervised machine learning approach and is composed of two steps: First, the process models to be matched are analysed to extract activities labels, next the syntactic category of each label is identified, then semantically related words are identified using WordNet. Second a king of local search is done to find the final alignment.

In (Xue, 2019), the author presented the compact memetic algorithm-based process model matching technique. This techniques tries to detect threshold that maximize the f-measure value for each matched pair of BPs.

THE PROPOSED METHOD

The proposed method for business process model matching is depicted in Figure 2. This method is composed of 4 steps: Activity extraction and Pre-processing, Activity labels embedding and Match detection.

Figure 2. The proposed method


PM1
PM2
Activity labels extarction &preprocessing
Activity labels embedings
Match detection
A set of correspondences

Activity Labels Extraction and Pre-Processing

This step extracts activity labels from the two process models to be aligned, pre-process them and discards meaningless words. It aims at transforming the activity label to a new format that help improving the result of the matching process. This activity consists of the following steps. First, a tokenisation step divides the activity label into words. Next, a normalisation step transforms each word in a lowercase form. Then, a noise removal step removes numbers, special characters and meaningless words. A word is meaningless if it does not influence the matching step. Example of such words are prepositions (e.g., are, in, and, or, there). Meaningless words are removed based on a stop word list.

Activity Label Embedding

Activity label embedding consists of representing an activity label previously pre-processed with a N-dimensional semantic vectors that represents its meaning.

This step builds this representation based on a word embedding model. a word embedding activity is a task that consist on representing words composing activities with numerical (i.e., real) vector. This vector represents the meaning of each word. The idea that support the proposed approach is based on the assumption that two words are semantically similar if they share the same properties (e.g., semantic similarity) with the other words. For example, if word W1 is similar to W2 and W3 and is different than W4. And if W5 is also close to W2 and W3 and different from W4 to some extent, then W1 is similar to W5.

To build such representation, each word composing the two activity labels is mapped to an M dimensional vector. Where M is the size of the vocabulary of the two activities labels (i.e., unique words in the two activities). Formally:

Let A_1 and A_2 two activity labels after pre-processing. The set of unique terms composing A_1, $UTA_1=\{T_{1a}...T_{nB}\}$ and the set of unique terms composing A_2, $UTA_2=\{T_{1b},..T_{mb}\}$. The vocabulary V= $UTA_1 \cup UTA_2$ is composed by unique terms found in A_1 and A_2.

A word W in A_1 or A_2 is represented (i.e., mapped to) with a real vector VW_1= { $w_1,w_2,...,w_s$}, s=|V| and w_i is the maximal score of similarity found between W and T_i. The stem of W and the stem of T_i and the lemma of W and the lemma of $T_{i.}$

A stem of a word W is derived by a process of stemming which consist on reducing inflected word to its root form by deleting affixes (e.g., eliminating suffix and/or prefix). For example, the stem of the word connected is connect. The lemma of a word W is found using a process of lemmatisation, a more sophisticated process than stemming, that consist of finding the root of the word W. For example the lemma of the word better is good.

The score of similarity is calculated using the Jiang–Conrath distance. This metric is a semantic similarity metric based on the WordNet thesaurus (Budanitsky & Hirst, 2006).This process is outlined by algorithm 1.

Based on these words representations, the vector representation of the activity A_i is created by averaging the vectors of words composing the activity A_i (Mitchell & Lapata, 2008).

Formally: let A_i be composed of the set of words $\{wr_1,wr_2,....,wr_s\}$ Where the vector representation of each word $wr_i=\{wwr_1,wwr_2,....,wwr_s\}$. The representation of A_i in a S dimensional space is:

$VA_i=\{WA_1,WA_2,.....,WA_s\}$, such that $WA_i=\dfrac{\sum_{i=1}^{i=s} wwri}{s}$. This vector represents the meaning of the activity A_i. The process used to compute the representation is outlined by algorithm 2.

Algorithm 1. Algorithm for embedding words composing activity labels

```
Inputs: two pre-processed activity Labels: $A_1$ and $A_2$
Output: a vector representation of each word in $A_1$ and $A_2$stored in a hash table.

L=tokenize($A_{1)}$U tokenize($A_2$) ;//L1 list containing words composing A1.
Remove duplicate words from(L1);
Initialize the hash table map to null;
// map stores words and their vector representations.
For each word $w_1$ in L do
{
initialize an empty array V of dimension |L|;
For each $w_2$ in L do {
Score=Max(Sim($w_1$,$w_2$),Sim(stem($w_1$),stem($w_2$)),Sim(lemma(w1),lemma(w2)));// sim calculate the similarity score between $W_1$ and $W_2$.
V.add(score);
}
map.add($w_1$, $_V$);
// add the word $w_1$ and its vector representation V to the hash table map.
}
```

Algorithm 2. Algorithm for embedding activity labels

```
Input: A pre-processed activity Label A, a map that store words representation.
Output: A vector representation of A.
L =tokenize(A);
Initialize VA to an empty vector of size=|A|;
For each word W in A
VA=VA+map.get(w);
//map.get(w) returns the vector representation of the word W from Map ;
VA=VA/|A|.
1.1
```

Match Detection

Given as an input two activity labels, this step calculates a score that indicates their degree of similarity.

A pair of activity labels A1 and A2 are reported as similar and consequently as a part of the correspondences composing the alignment if their similarity score computed using the cosine metric is greater than some predefined threshold. Formally, let VA_1= {WA_{11}, WA_{12},..,WA_1s} and VA_2={WA_{21},WA_{22},.....,WA_{2s}} be the two vector representation of A1 and A2 respectively. The semantic similarity is computed using the following formula:

$$\text{Sim } (VA_1, VA_2) = \frac{VA_1 * VA_2}{VA_1 * VA_2} .$$

The semantic similarity is computed only in the case where the two activities are syntactically dissimilar. The syntactic similarity is computed using the Levenshtein distance (Levenshtein, 1960).

EMPIRICAL STUDY

Design

To conduct the empirical study, a tool named BPMWE (Business Process Matching based on Word Embedding) was developed. This tool implements the proposed method presented in Figure 2 and it is developed using the JAVA language. The empirical study was conducted to assess the performance of the proposed method. The empirical study consists of two experiments. The main hypothesis is that our method is appropriate for matching process model. More specifically, the empirical study intends to answer the following research question.

RQ1: Process model matching Performance: How is the precision and recall of the proposed method compared to other techniques?

In order to answer RQ1, an experiment was executed on three well known datasets (Goncalo et al. 2015). Then three widely used measures by the process management community are used to evaluate the proposed method. These measures are Precision, Recall and F-measure.

Finally, the results of the experiments are compared to the results of state-of-the-art matchers to draw a final conclusion.

The Datasets

The empirical study used three well-known sets of process models (Goncalo et al. 2015):

1. University Admission Processes (UA): This set consists of 36 model pairs that were derived from 9 models representing the application procedure for Master students of nine German universities.
2. Birth Registration Processes (BR): This set consists of 36 model pairs that were derived from 9 models representing the birth registration processes of Germany, Russia, South Africa, and the Netherlands.
3. Asset Management (AM): This set consists of 36 model pairs that were derived from 72 models from the SAP Reference Model Collection.

These datasets come with an oracle (i.e., golden set) containing information about matches existing between each pair of two process models. Based on information contained in this oracle , each computed by our method is classified as either true-positive (TP), true-negative (TN), false-positive (FP) or false-negative (FN). Then, Precision, Recall and f-measure are calculated using the following formulas:

$$\text{Precision}=\frac{TP}{TP+FP}.$$

$$\text{Recall}=\frac{TP}{TP+FN}.$$

$$\text{F-measure}=\frac{2*precision*recall}{precision+recall}$$

Results

Table 1 shows the Precision, the Recall and F-measure values obtained by executing the BPMWE matcher on the three UA, BR and AM. These are the best values obtained when setting the threshold to 0.98, 0.77 and 0.98 in the case of the UA, BR and AM datasets respectively.

Table 1. Precision, Recall and F-measure values of BMWE method

	UA	BR	AM
Precision	0,847	0,499	0,956
Recall	0,478	0,446	0,486
F-measure	0,611	0,471	0,644

The Figure 3 shows that the precision values achieved by the BPMWE was 0.847,0.499 and 0.956 in the case of UA,BR and AM datasets and compare it to the precision the state-of-the-art matchers. These values was very high.

The Figure 4 shows that the Recall values achieved by the BPMWE was 0.478, 0.446 and 0.486 in the case of UA, BR and AM datasets and compare it to the Recall the state-of-the-art matchers. The Figure 4 shows that the values were balanced in all cases.

The Figure 5 shows that the F-measure values achieved by the BPMWE was 0.611, 0.471 and 0.644 in the case of UA, BR and AM datasets and compare it to the F-measure the state of the art matchers.

Figure 3. Precision of the BMWE method

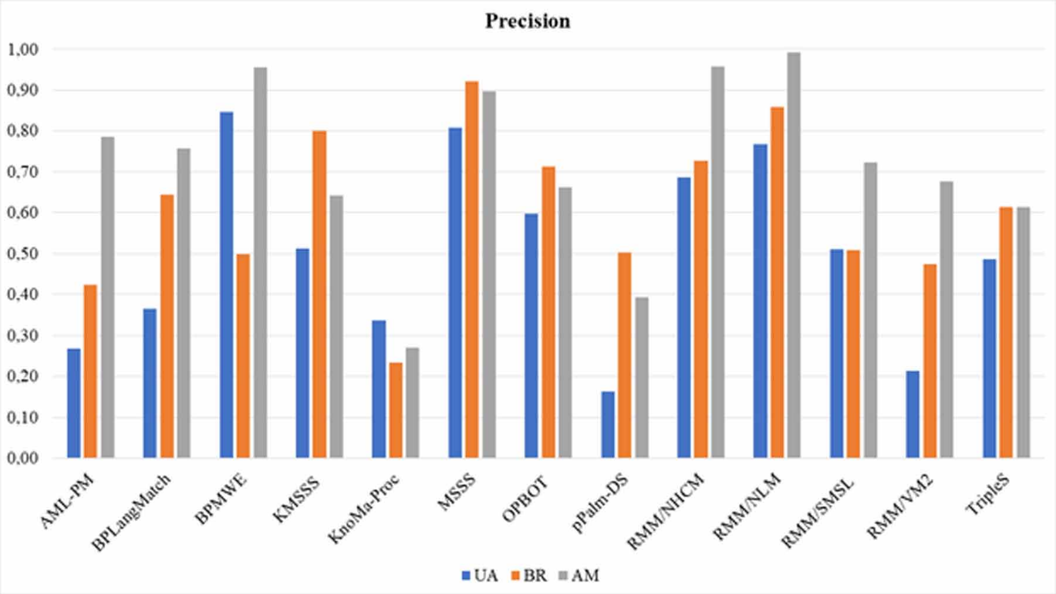


Figure 4. The Recall of the BMWE method

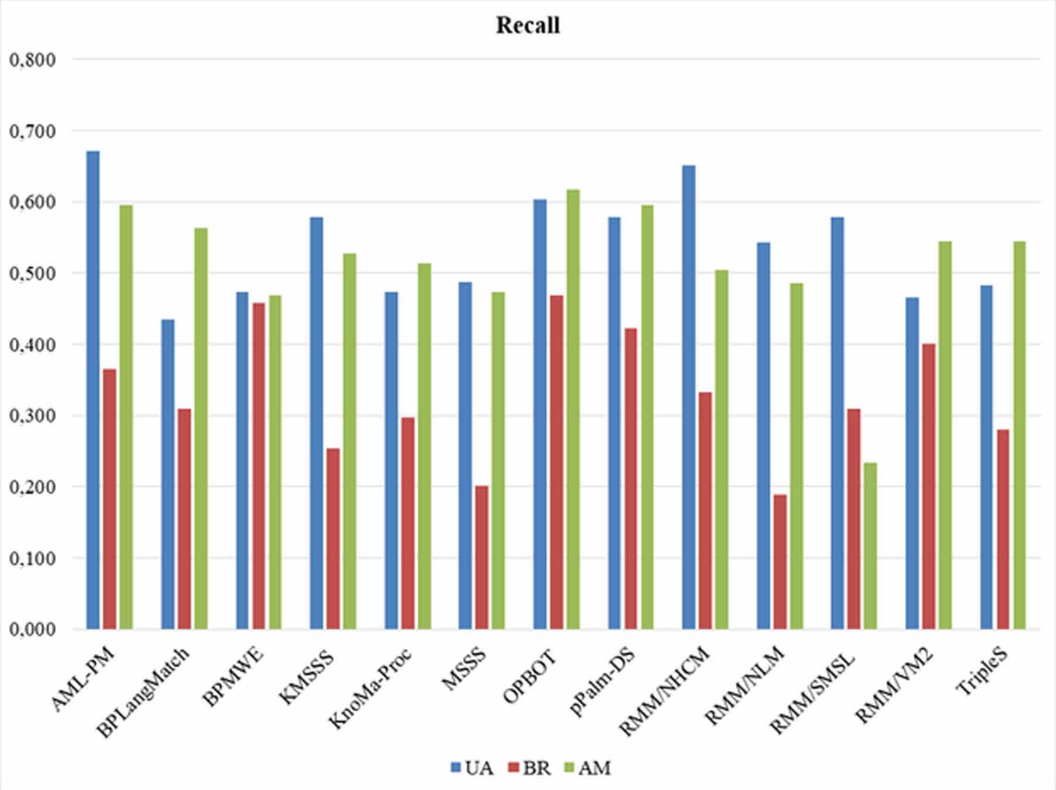
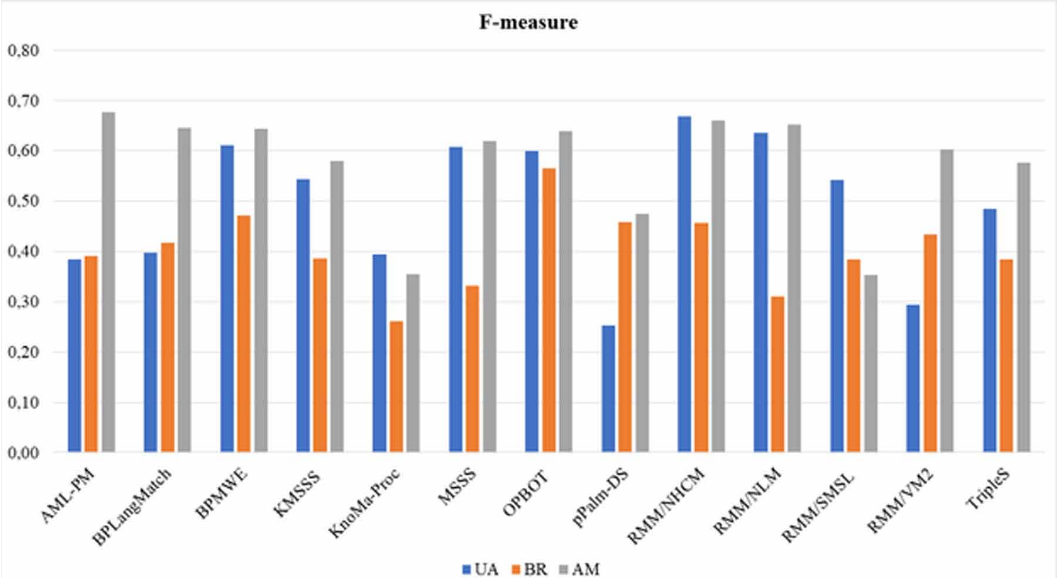


Figure 5. The F-measure of the BMWE method



To study the quality of the obtained results by the proposed matcher, these results need to be compared with the state-of-the-art matchers.

The Table 2 compares the f-measure values of the proposed matcher to f-measure values obtained by state-of-the-art matchers.

From this table and from Figures 3, 4 and 5, It is very important to note that no matcher was able to outperform all the others in all cases.

The table shows that matchers that outperformed all other matcher in one case are less effective in other cases.

For example, AML-PM outperformed all the other in the case of the AM dataset but it was less effective in the other two cases. Its ranks were 10 in the case of UA dataset and 6 in the case of BR dataset. The proposed method, BPMWE, outperformed AML-PM in two cases (UA and BR) from three.

The RMM/NHCM outperformed all the other in the case of the UA dataset but it was less effective in the other two cases. Its rank was 2 in the case of AM dataset and 4 in the case of BR dataset. The proposed method, BPMWE, outperformed RMM/NHCM in one case (BR) from three.

The OPBOT outperformed all the other in the case of the BR dataset but it was less effective in the other two cases. Its rank was 5 in the case of UA dataset and 6 in the case of AM dataset. The proposed method, BPMWE, outperformed OPBOT in two cases (UA and AM) from three.

This table indicates also that the rank of the proposed method (BPMWE) was 3 after RMM/NHCM and RMM/NLM, 2 after OPBOT and 5 after AML-PM, RMM/NHCM, RMM/NLM and BPLangMatch in the case of UA, BR and AM datasets respectively.

These information show that it is very difficult to calculate an exact rank for our approach or for any other matcher. But it indicates clearly that the proposed method share the first place with RMM/NHCM since the two approaches outperformed all the other matchers in two cases from three.

Thus the proposed method BPMWE is effective than 11 matchers and shares the first place with RMM/NHCM.

To compare quantitatively BPMWE with other matchers the average F-measure score achieved by each matcher was calculated. These scores are presented in Table 3.

Table 2. A comparison of F-measure values of the state-of-the-art matchers to the F-measure values of BMWE

F-measure	UA	BR	AM
OPBOT	0,600	0,565	0,639
RMM/NHCM	0,668	0,457	0,661
BPMWE	0,611	0,471	0,644
AML-PM	0,384	0,392	0,677
KMSSS	0,544	0,386	0,579
BPLangMatch	0,397	0,418	0,646
KnoMa-Proc	0,394	0,262	0,355
MSSS	0,607	0,331	0,619
pPalm-DS	0,253	0,459	0,474
RMM/NLM	0,636	0,310	0,652
RMM/SMSL	0,542	0,384	0,353
RMM/VM2	0,293	0,434	0,603
TripleS	0,485	0,384	0,577

This Table confirms the previous conclusion and shows that the BPMWE rank is 3 after OPBOT and RMM/NHCM. Despite that OPTBOT was outperformed by the proposed method BPMWE and RMM/NHCM in two cases, in average it is ranked first because of the high value obtained in the case of BR where it outperformed BPMWE and RMM/NHCM.

Consequently, the answer to the research question RQ1 is given as follow:

The proposed method shared the first place with RMM/NHCM and it have the potential to match BPs effectively.

Table 3. A comparison of the average F-measure values of the state-of-the-art matchers to the average F-measure values of BMWE

	Average F-measure
OPBOT	0,601
RMM/NHCM	0,595
BPMWE	0,576
RMM/NLM	0,533
MSSS	0,519
KMSSS	0,503
AML-PM	0,484
BPLangMatch	0,487
TripleS	0,482
RMM/VM2	0,444
RMM/SMSL	0,427
pPalm-DS	0,395
KnoMa-Proc	0,337

CONCLUSION

This paper proposed a new method for process model matching and a tool named BPMWE. The proposed method detects semantically equivalent activities based on a new word embedding approach. First activity labels to be compared are extracted from process models and pre-processed to remove meaningless words, then each word composing an activity label is represented with an N-dimensional vector in the space of the vocabulary of the two labels and using WordNet. Based on this representation each activity label is represented with N-dimensional vector by averaging the vectors representing its composing words. Finally, the two activity labels are reported as similar if their similarity score computed using the cosine metric is greater than some predefined threshold. An experiment was conducted on well-known datasets to assess the performance of the reposed method. The results showed that our method outperformed most of the matchers proposed in the literature. The results show that the method shared the first place with RMM/NHCM and OPBOT.

A future work will be dedicated to improve the effectiveness of the proposed approach and to evaluate it in different domains that require matching short texts such as ontology matching and NLP domains.

REFERENCES

Agirre, Banea, Cer, Diab, Gonzalez-Agirre, Mihalcea, Rigau, & Wiebe. (2016). SemEval-2016 Task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. *Proceedings of the Workshop on Semantic Evaluation*, 497–511.

Antunes, G., Bakhshandeh, M., Borbinha, J., Cardoso, J., Dadashnia, S., Di Francescomarino, C., Dragoni, M., & Fettke, P. (2015). The Process Model Matching Contest 2015. *6th EMISA Workshop,* 127–155.

Beheshti, S.-M.-R., Benatallah, B., Sakr, S., Grigori, D., Motahari-Nezhad, H. R., Barukh, M. C., Gater, A., & Ryu, S. H. (2006). *Process Analytics Concepts and Techniques for Querying and Analyzing Process Data*. Springer International Publishing AG.

Budanitsky, A., & Hirst, G. (2006). Evaluating WordNet-based measures of semantic distance. *Computational Linguistics*, *32*(1), 13–47. doi:10.1162/coli.2006.32.1.13

Cayoglu, U., Dijkman, R., Dumas, M., Fettke, P., Garcıa-Banuelos, L., Hake, P., Klinkm¨uller, C., Leopold, H., Ludwig, A., & Loos, P. (2013) The process model matching contest 2013. *4th International Workshop on Process Model Collections: Management and Reuse (PMC-MR'13).*

Cayoglu, U., Oberweis, A., Schoknecht, A., & Ullrich, M. (2013). *Triple-S: Matching Approach for Petri Nets on Syntactic, Semantic and Structural level.* Technical report.

Christopher, K., Henrik, L., Ingo, W., Mendling, J., & Ludwig, A. (2014). Listen to Me: Improving Process Model Matching through User Feedback. Business Process Management, 84–100.

Euzenat, J., & Shvaiko, P. (2007). *Ontology matching*. Springer-Verlag.

Faria, D., Pesquita, C., Santos, E., Cruz, I. F., & Couto, F. M. (2013) AgreementMakerLight results for OAEI 2013. OM, 101–108.

Gal, A., & Sagi, T. (2010). Tuning the ensemble selection process of schema matchers. *Information Systems*, *35*(8), 845–859. doi:10.1016/j.is.2010.04.003

Gomaa, W. H., & Fahmy, A. A. (2013). A survey of text similarity approaches. *International Journal of Computers and Applications*, *68*(13), 13–18. doi:10.5120/11638-7118

Jabeen, F., Leopold, H., & Reijers, H. A. (2017) How to Make Process Model Matching Work Better? An Analysis of Current Similarity Measures. *International Conference on Business Information Systems*.

Jin, T., Wang, J., La Rosa, M., Ter Hofstede, A., & Wen, L. (2013). Efficient querying of large process model repositories. *Computers in Industry*, *64*(1), 41–49. doi:10.1016/j.compind.2012.09.008

La Rosa, M., Dumas, M., Uba, R., & Dijkman, R.M. (2013). An approach to business process consolidation. *ACM Transactions on Software Engineering and Methodology*, *22*(2), 11. doi:10.1145/2430545.2430547

Levenshtein, V. I. (1960). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics, Doklady*, *10*(8), 707–710.

Meilicke, C., Leopold, H., Kuss, E., Stuckenschmidt, H., & Reijers, H. A. (2017). Overcoming individual process model matcher weaknesses using ensemble matching. *Decision Support Systems*, *100*, 15–26. doi:10.1016/j.dss.2017.02.013

Mendling, J., Reijers, H. A., & Recker, J. (2010). Empirical Insights and Recommendations. *Information Systems*, *35*(4), 467–482. doi:10.1016/j.is.2009.03.009

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, *38*(11), 39–41. doi:10.1145/219717.219748

Miller, G. A., Beckitch, R., Fellbaum, C., Gross, D., & Miller, K. (1993). *Introduction toWordNet: An On-line Lexical Database*. Princeton University Press.

Mitchell, J., & Lapata, M. (2008). Vector-based models of semantic composition. *Proceedings of ACL*, 236–244.

Rahm, E., & Bernstein, P. A. (2001). A survey of approaches to automatic schema matching. *The VLDB Journal*, *10*(4), 334–350. doi:10.1007/s007780100057

Shahzad, K., Kanwal, S., Malik, K., Aslam, F., & Ali, M. (2019, September). A word-embedding-based approach for accurate identification of corresponding activities. *Computers & Electrical Engineering*, *78*, 218–229. doi:10.1016/j.compeleceng.2019.07.011

Wang, J., Jin, T., Wong, R. K., & Wen, L. (2014). Querying business process model repositories - Asurvey of current approaches and issues. *World Wide Web (Bussum)*, *17*(3), 427–454. doi:10.1007/s11280-013-0210-z

Weidlich, M., Dijkman, R. M., & Mendling, J. (2010). The ICoP framework: Identification of correspondences between process models. CAiSE, 483-498.

Weidlich, M., Mendling, J., & Weske, M. (2012). Propagating changes between aligned process models. *Journal of Systems and Software*, *85*(8), 1885–1898. doi:10.1016/j.jss.2012.02.044

Weidlich, M., Sheetrit, E., Branco, M., & Gal, A. (2013) Matching Business Process Models Using Positional Language Models. *32nd International Conference on Conceptual Modeling, ER.*

Xue, X. (2019). Compact memetic algorithm-based process model matching. *Soft Computing*, *23*(13), 5249–5257. doi:10.1007/s00500-018-03672-y

Yates, R. B., & Neto, B. R. (1999). *Modern Information Retrieval*. Addison-Wesley.

Mostefai Abdelkader (PhD) is an assistant professor in the Department of Computer Science of Dr Tahar Moulay university of Saida, Algeria. His work focuses on software evolution, Artificial intelligence, Data mining and Business Process Management.

Mansour Mekour is an Associate Professor at the Computer Science Department of Saida University, Algeria. He received his PhD in Computer Science from Sidi Bel Abbes University, Algeria, in 2014. He is currently Membership of Research Team 'Service Oriented Computing' at the LabRI Laboratory, Ecole Superieure en Informatique, Sidi Bel Abbes, Algeria. His main research interests include Semantic Web, service-oriented computing, web services, QoS, and web interaction.