Similarity of Business Process Models: Metrics and Evaluation

Anna E. Derguzova
Software Engineering Department
Repositories of business process models serve as a knowledge base for business process management efforts.

- **Hundreds** of business process models
- **Thousands** of business process models
Business Process Models Repositories - Examples

Repository of Dutch local government council – ~500 process models

SAP reference model repository – ~600 process models
Business Process Models Repositories - Search

The management and use of large process model repositories requires effective *search techniques*

- to check that a similar model does not already exist in order to prevent duplication

- to identify common or similar business processes in order to analyze their overlap and to identify areas for consolidation
Need to retrieve process models based on their similarity with respect to a given “search model”.

The term **process model similarity query** is used to refer to such search queries over process model repositories.
Traditional **search engines** can be used to index and to search business process model repositories.

Search engines are:

- based on *keyword search* and *text similarity*

- useful in situations where a user is looking for a model that contains an activity with a certain *keyword in its label*

- **but** hardly appropriate for process model similarity queries, since they do not take into account the *structure and behavioral semantics* of process models.
Similarity Metrics

Similarity metrics to answer process model similarity queries

- LABEL matching similarity
- STRUCTURAL similarity
- BEHAVIORAL similarity
Numerous notations compete in the business process modeling space:

- UML Activity Diagrams
- Business Process Modeling Notation (BPMN)
- Workflow nets
- Business Process Execution Language (BPEL)
- Event-driven Process Chains (EPCs)
- ...
The **EPC** notation - a graph-based language for documenting the temporal and logical dependencies between functions and events in an organization.

**Definition 1 (EPC).** An EPC is a tuple \((F, E, C, l, A)\) and \(\Omega\) a set of text labels, in which:
- \(F\) is a finite set of functions;
- \(E\) is a finite set of events;
- \(C\) is a finite set of connectors;
- \(l : (F \cup E \rightarrow \Omega) \cup (C \rightarrow \{\text{and}, \text{xor}, \text{or}\})\) labels functions and events with text and connectors with types;
- \(A \subseteq (F \cup E \cup C) \times (F \cup E \cup C)\) is the set of arcs.

An EPC is syntactically correct if and only if it contains at least one function and has strict alternation of events and functions on each path of arcs from start to end.

![Fig. 1. Example EPC](image-url)
Notations and Notions.

Process Modeling and EPCs

Notions of **path** and **connector chain** are used to discuss the relations between events and functions.

**Definition 2** (Paths and Connector Chains). Let $(F, E, C, I, A)$ be an EPC and $N = F \cup E \cup C$ be the set of nodes of that EPC. For each node $n \in N$, we define path $a \leftrightarrow b$ refers to the existence of a sequence of EPC nodes $n_1, \ldots, n_k \in N$ with $a = n_1$ and $b = n_k$ such that for all $i \in 1, \ldots, k$ holds: $(n_i, n_{i+1}) \in A$. This includes the empty path of length zero, i.e., for any node $a : a \leftrightarrow a$. Let $M \subseteq N$ be a set of nodes and $a \neq b \in N$ be two nodes. A path containing only $n_2, \ldots, n_{k-1} \in M$, denoted $a \overset{M}{\leftrightarrow} b$ is called a restricted path. This includes the empty restricted path, i.e., $a \overset{M}{\leftrightarrow} b$ if $(a, b) \in A$. The path restricted to the set of connectors, denoted $a \overset{C}{\leftrightarrow} b$, is called a connector chain.
**Notations and Notions.**

**Causal Footprint of an EPC**

**Causality graph** - a set of activities and conditions on when those activities can occur.

*Causality graph* is a formal semantics that approximates the behavior of a business process, in which case we also refer to it as the *causal footprint* of that process.

*Causality graph* represents behavior between a set of activities by means of two relationships: *look-back* and *look-ahead links*.

*Look-ahead link* from an activity to a (non-empty) set of activities: execution of the activity leads to the execution of at least one of the activities in the set.

\((a, B)\) - *look-ahead link* → any execution of \(a\) is followed by the execution of some \(b \in B\).

*Look-back link* from a (non-empty) set of activities to an activity: execution of the activity is preceded by the execution of at least one of the activities in the set.

\((A, b)\) - *look-back link* → any execution of \(b\) is preceded by the execution of some \(a \in A\).
Notations and Notions.

Causal Footprint of an EPC

**Definition 3 (Causality Graph).** A causality graph is a tuple \((A, L_{lb}, L_{la})\), in which:

- \(A\) is a finite set of activities;
- \(L_{lb} \subseteq (\mathcal{P}(A) \times A)\) is a set of look-back links\(^1\);
- \(L_{la} \subseteq (A \times \mathcal{P}(A))\) is a set of look-ahead links.

A causality graph is a causal footprint of an EPC if and only if it is consistent with the behavior of that EPC.

**Definition 4 (Causal Footprint of an EPC).** Let \(P = (F, E, C, l, A)\) be an EPC, \(G = (F, L_{lb}, L_{la})\) be a causality graph over the functions of \(P\), and \(W \subseteq F^*\) be the set of possible orders in which functions from \(P\) can be performed. \(G\) is a causal footprint of \(PC\) if and only if:

1) For all \((a, B) \in L_{la}\) holds that for each \(\sigma \in W\) with \(n = |\sigma|\), such that there is a \(0 \leq i \leq n-1\) with \(\sigma[i] = a\), there is a \(j : i < j \leq n - 1\), such that \(\sigma[j] \in B\),

2) For all \((A, b) \in L_{lb}\) holds that for each \(\sigma \in W\) with \(n = |\sigma|\), such that there is a \(0 \leq i \leq n - 1\) with \(\sigma[i] = b\), there is a \(j : 0 \leq j < i\), such that \(\sigma[j] \in A\).
Notations and Notions.

Causal Footprint of an EPC

Example:

Possible causal footprint for the EPC from figure 1 has

*look-ahead link:*

(‘Receive Goods’, {‘Verify Invoice’, ‘Transfer to Warehouse’})

and *look-back links:*

({‘Receive Goods’}, ‘Verify Invoice’)
({‘Receive Goods’}, ‘Transfer to Warehouse’)

Fig. 1. Example EPC
Fig. 2. Customer Inquiry and Customer Inquiry and Quotation Processing EPCs.
Notations and Notions.

Similarity of Process Model Elements

Ways of measuring similarity between elements of different process models

SYNTACTIC similarity

SEMANTIC similarity

CONTEXTUAL similarity
SYNTACTIC similarity

Given two labels, the syntactic similarity metrics returns the degree of similarity as measured by the *string-edit distance*.

*String-edit distance* - the **number of atomic string operations** necessary to get from one string to another.

**Atomic string operations:**
- removing a character
- inserting a character
- substituting a character for another

**Definition 5 (Syntactic similarity).** Let \((E_1, F_1, C_1, l_1, A_1)\) and \((E_2, F_2, C_2, l_2, A_2)\) be two disjoint EPCs. Furthermore let \(n_1 \in F_1 \cup E_1 \cup C_1\) and \(n_2 \in F_2 \cup E_2 \cup C_2\) be two nodes from those EPCs and let \(l_1(n_1)\) and \(l_2(n_2)\) be the two strings that represent the labels of those nodes, i.e. we can calculate their length, denoted \(|l_1(n_1)|\) and \(|l_2(n_2)|\), and their edit distance, denoted \(ed(l_1(n_1), l_2(n_2))\). We define the syntactic similarity of EPC nodes \(n_1\) and \(n_2\) as follows:

\[
syn(n_1, n_2) = 1 - \frac{ed(l_1(n_1), l_2(n_2))}{\max(|l_1(n_1)|, |l_2(n_2)|)}
\]
SYNTACTIC similarity

Example:

Syntactic similarity between the events $e_{12}$ and $e_{21}$ from figure 2 with labels “Customer inquiry about product” and “Customer inquiries about product” is

$$1 - \frac{3}{30} = 0.90$$

because the edit distance is 3:

“inquiries” becomes “inquiry” by substituting ‘y’ with ‘i’ and inserting ‘e’ and ‘s’.
SEMANTIC similarity

Given two labels, the semantic similarity score is the degree of similarity based on equivalence between the words they consist of.

Exact match is preferred over a match on synonyms:

Identical words - equivalence score of 1.

Synonymous words - equivalence score of 0.75.

**Definition 6 (Semantic similarity).** Let \((E_1, F_1, C_1, l_1, A_1)\) and \((E_2, F_2, C_2, l_2, A_2)\) be two disjoint EPCs. Furthermore let \(n_1 \in F_1 \cup E_1 \cup C_1\) and \(n_2 \in F_2 \cup E_2 \cup C_2\) be two nodes from those EPCs and let \(w_1 = l_1(n_1)\) and \(w_2 = l_2(n_2)\) be the labels of those nodes (and assume that \(w_1\) and \(w_2\) are sets of words, i.e. we denote the number of words by \(|w_1|\) and \(|w_2|\) and we can use standard set operators). We define the semantic similarity of EPC nodes \(n_1\) and \(n_2\) as follows:

\[
\text{sem}(n_1, n_2) = \frac{1.0 \cdot |w_1 \cap w_2| + 0.75 \cdot \sum_{s \in w_1 \setminus w_2} \text{synonym}(s, t)}{\max(|w_1|, |w_2|)}
\]

Where synonym is a function that returns 1 if the given words are synonyms and 0 if they are not.
Notations and Notions.

Similarity of Process Model Elements

**SEMANTIC similarity**

**Example:**

Consider the functions $f_{11}$ and $f_{21}$ from figure 2 with labels “Customer inquiry processing” and “Client inquiry query processing”.

Labels consist of the collections of words:

$w_1 = [\text{"Customer"}, \text{"inquiry"}, \text{"processing"}]$

$w_2 = [\text{"Client"}, \text{"inquiry"}, \text{"query"}, \text{"processing"}]$

We only need to consider a synonym mapping between

$w_1 \setminus w_2 = [\text{"Customer"}]$ and $w_2 \setminus w_1 = [\text{"Client"}, \text{"query"}]$

“Customer” and “Client” are synonymous and “Customer” and “query” are not.

**Semantic similarity** between $w_1$ and $w_2$: 

$$sem(w_1, w_2) = \frac{1.0 \cdot 2 + 0.75 \cdot (1 + 0)}{4} \approx 0.69$$
Notations and Notions.

Similarity of Process Model Elements

SEMANTIC similarity

Fig. 2. Customer Inquiry and Customer Inquiry and Quotation Processing EPCs.
CONTEXTUAL similarity

When comparing two functions, the *contextual similarity metric* takes the surrounding events into account: in EPCs functions are always preceded and succeeded by events.

Preceding model elements – *input context*
Succeeding model elements – *output context* of another model element.

**Definition 7** (Input and output context). Let \((E, F, C, l, A)\) be an EPC. For a node \(n \in F \cup E\), we define the input context \(n^\text{in} = \{n' \in F \cup E \mid n' \xleftarrow{C} n\}\) and the output context \(n^\text{out} = \{n' \in F \cup E \mid n \xrightarrow{C} n'\}\).
**Definition 8 (Equivalence Mapping).** Let $L_1$, $L_2$ be two disjoint sets. Furthermore, let $s : L_1 \times L_2 \rightarrow [0..1]$ be a similarity function such that for all $l_1 \in L_1$ and $l_2 \in L_2$: $s(l_1, l_2) = s(l_2, l_1)$. A partial injective mapping $M_s : L_1 \rightarrow L_2$ is an equivalence mapping, if and only if for all $l_1 \in L_1$ and $l_2 \in L_2$: $M(l_1) = l_2$ implies that $s(l_1, l_2) > 0$.

An optimal equivalence mapping $M_s^{opt} : L_1 \rightarrow L_2$ is an equivalence mapping, such that for all other equivalence mappings $M$ holds that

$$\sum_{(l_1, l_2) \in M_s^{opt}} s(l_1, l_2) \geq \sum_{(l_1, l_2) \in M_s} s(l_1, l_2).$$

**Example:**
Consider equivalence mapping between \{e12\} and \{e21, e22\}. Assume *syntactic similarity* ($syn$) as a similarity function.

$M_{syn} = \{(e12, e22)\}$ - possible equivalence mapping, because $syn(e12, e22) \approx 0.24$.

$M_{optsyn} = \{(e12, e21)\}$ - optimal equivalence mapping, because $syn(e12, e21) = 0.90$. The only other possible mapping is the empty mapping.
Notations and Notions.

Similarity of Process Model Elements

CONTEXTUAL similarity

Fig. 2. Customer Inquiry and Customer Inquiry and Quotation Processing EPCs.
Notations and Notions.

Similarity of Process Model Elements

**Definition 9 (Contextual Similarity).** Let \((E_1, F_1, C_1, l_1, A_1)\) and \((E_2, F_2, C_2, l_2, A_2)\) be two disjoint EPCs. Let \(n_1 \in F_1\) and \(n_2 \in F_2\) be two functions and let Sim be one of the similarity functions.

Furthermore, let \(M_{Sim}^{optin} : n_1^{in} \rightarrow n_2^{in}\) and \(M_{Sim}^{optout} : n_1^{out} \rightarrow b_2^{out}\) be two optimal equivalence mappings between the input and output contexts of \(n_1\) and \(n_2\) respectively. We define the contextual similarity as follows:

\[
\text{con}(n_1, n_2) = \frac{|M_{Sim}^{optin}|}{2 \cdot \sqrt{|n_1^{in}| \cdot |n_2^{in}|}} + \frac{|M_{Sim}^{optout}|}{2 \cdot \sqrt{|n_1^{out}| \cdot |n_2^{out}|}}
\]
**Label matching similarity** is based on pairwise comparisons of node labels.

**Label matching similarity** is obtained by calculating an *optimal equivalence mapping* between the nodes of the two process models being compared.

**Definition 10** (Label Matching Similarity). Let $P_1 = (F_1, E_1, C_1, l_1, A_1)$ and $P_2 = (F_2, E_2, C_2, l_2, A_2)$ be two EPCs and let $\text{Sim}$ be a function that assigns a similarity score to a pair of functions/events. Let $M_{\text{Sim}}^{\text{opt}} : (F_1 \rightarrow F_2) \cup (E_1 \rightarrow E_2)$ be an optimal equivalence mapping derived from $\text{Sim}$. The label matching similarity between $P_1$ and $P_2$ is:

$$\text{simlmbm}(P_1, P_2) = \frac{2 \cdot \sum_{(n,m) \in M_{\text{Sim}}^{\text{opt}}} \text{Sim}(n, m)}{|F_1| + |F_2| + |E_1| + |E_2|}$$
Parameterize the label matching similarity metrics with a threshold between 0 and 1.

When calculating an optimal equivalence mapping, we only allow two nodes to be included in the equivalence mapping if their similarity is above the threshold.

With respect to Definition 8, this means that instead of enforcing that \( s(l_1, l_2) > 0 \), we enforce that \( s(l_1, l_2) \geq \text{threshold} \).
Example:

The \textit{optimal equivalence mapping} between EPCs from figure 2 is denoted by the two-way arrows with the $=$ symbol.

Assuming that we use \textit{syntactic equivalence (syn)} to determine the similarity between the functions and events, and that we use a \textit{threshold of 0.5}, the \textit{similarity score} of the elements included in the equivalence mapping is:

\begin{align*}
\text{syn}(e_{12}, e_{21}) &= 0.90 \\
\text{syn}(f_{11}, f_{21}) &\approx 0.58 \\
\text{syn}(e_{13}, e_{23}) &= 1.00
\end{align*}

The remaining elements are not included in the equivalence mapping because the syntactic similarity score between all other possible pairs of elements is less than 0.5.

Hence, the \textit{label matching similarity} between these two EPCs is:

\[
\frac{2 \cdot \sum_{(n,m) \in M_{\text{syn}}^{\text{opt}}} \text{syn}(n, m)}{|F_1| + |F_2| + |E_1| + |E_2|} = \frac{2 \cdot (0.90 + 0.58 + 1.00)}{3 + 1 + 4 + 6} \approx 0.35
\]
Consider EPC as a labeled graph:

**EPC:**
- Functions
- Events
- Connectors
- Arcs

**Graph:**
- Nodes
- Edges

Labels of Functions and Events

Types of Connectors (*and*, *or*, *xor*)
**Structural similarity score** of two EPCs = **graph-edit distance**.

The **graph-edit distance** between two graphs is the **minimal number of graph-edit operations** that is necessary to get from one graph to the other.

**Graph-edit operations:**
- node deletion or insertion
- node substitution (a node in a graph is mapped to a node in the other graph with a different label)
- edge deletion or insertion
Definition 11 (Graph Edit Distance). Let $P_1 = (F_1, E_1, C_1, l_1, A_1)$ and $P_2 = (F_2, E_2, C_2, l_2, A_2)$ be two EPCs. Let $N_1 = F_1 \cup E_1 \cup C_1$ be the nodes of $P_1$ and $N_2 = F_2 \cup E_2 \cup C_2$ be the nodes of $P_2$ and let $\text{Sim}$ be one of the similarity metrics from subsection 2.3. Let $M : (F_1 \rightarrow F_2) \cup (E_1 \rightarrow E_2) \cup (C_1 \rightarrow C_2)$ be a partial injective mapping that maps functions, events and connectors.

Let $n \in N_1 \cup N_2$ be a node. $n$ is substituted if and only if $n \in \text{dom}(M)$ or $n \in \text{cod}(M)$. $\text{sb}$ is the set of all substituted nodes. $n$ is inserted or deleted if and only if it is not substituted. $\text{sn}$ is the set of all inserted and deleted nodes.

Let $(n, m) \in A_1$ be an edge. $(n, m)$ is inserted in or deleted from $P_1$ if and only if there do not exist mappings $(n, n') \in M$ and $(m, m') \in M$ and edge $(n', m') \in A_2$. Edges that are inserted in or deleted from $P_2$ are defined similarly. $\text{se}$ is the set of all inserted or deleted edges.

The distance induced by the mapping is defined as:

$$|\text{sn}| + |\text{se}| + 2 \cdot \sum_{(n, m) \in M} 1 - (\text{Sim}(n, m))$$

The graph edit distance is the minimal possible distance induced by a mapping between the two processes.
Example:

Consider the EPCs from figure 2. Assume *syntactic equivalence* (*syn*) to determine the similarity between functions and events. The **structural similarity score** is:  

$$ 12 + 16 + 2 \cdot (1 - 0.90 + 1 - 0.58 + 1 - 1.00) \approx 29.04 $$

---

*Fig. 2. Customer Inquiry and Customer Inquiry and Quotation Processing EPCs.*
**Definition 12** (Graph Edit Distance Similarity). Let $P_1 = (F_1, E_1, C_1, l_1, A_1)$ and $P_2 = (F_2, E_2, C_2, l_2, A_2)$ be two EPCs. Let $N_1 = F_1 \cup E_1 \cup C_1$ be the nodes of $P_1$ and $N_2 = F_2 \cup E_2 \cup C_2$ be the nodes of $P_2$ and let $\text{Sim}$ be one of the similarity metrics.

Furthermore, let $M : (F_1 \leftrightarrow F_2) \cup (E_1 \leftrightarrow E_2) \cup (C_1 \leftrightarrow C_2)$ be a mapping that induces the graph edit distance between the two processes and let $\text{sn}$ and $\text{se}$ be defined as in definition 11. We define the graph edit distance similarity as:

$$\text{sim}_{\text{ged}}(P_1, P_2) = 1 - \{\text{snv, sev, sbv}\}$$

Where:

$$\text{snv} = \frac{|\text{sn}|}{|N_1| + |N_2|}$$

$$\text{sev} = \frac{|A_1| + |A_2|}{2|\Sigma_{(n, m) \in M} 1 - \text{Sim}(n, m)|}$$

$$\text{sbv} = \frac{|\text{sn}|}{|N_1| + |N_2| - |\text{sn}|}$$

**Definition 13** (Node abstraction). Let $P = (F, E, C, l, A)$ be an EPC, let $N = F \cup E \cup C$ be its nodes, $\Omega$ be the set of all possible labels and let $I \subseteq N$ be the subset of nodes to ignore. The EPC in which the nodes from $I$ are ignored is the EPC $P' = (F - I, E - I, C - I, l - (I \times \Omega), A')$, where $A' = \{(a, b) | a, b \in (N - I), a \overset{I}{\leftrightarrow} b, a \neq b\}$. 
Fig. 2. Customer Inquiry and Customer Inquiry and Quotation Processing EPCs.
**Behavioral similarity score** of two EPCs = *distance in the document vector space* that can be constructed from their *causal footprints*.

*Document vector space* model is used of the causal footprints of the EPCs, rather than of the EPCs themselves, to incorporate an approximation of behavior in the similarity metric.

*Document vector space* consists of:
- a collection of *documents* (two EPCs in our case)
- a set of *index terms* according to which the documents are indexed
- an *index vector* for each document that assigns a weight to each index term
Index terms

Index terms are derived from the sets of functions, look-ahead links and look-back links of the causal footprints.

Example:
Function labels “enter client information” and “enter client’s information” differ with respect to their labels, but could still be considered the same function.

Definition 14. Let $P_1$ and $P_2$ be two EPCs with causal footprints $G_1 = (F_1, L_{lb,1}, L_{la,1})$ and $G_2 = (F_2, L_{lb,2}, L_{la,2})$ and let $M : F_1 \rightarrow F_2$ be a partial injective mapping that associates similar functions. We define the set of index terms as: $\Theta = M \cup (F_1 - \text{dom}(M)) \cup L_{lb,1} \cup L_{la,1} \cup (F_2 - \text{cod}(M)) \cup L_{lb,2} \cup L_{la,2}$. In the remainder we consider the sequence of index terms $\lambda|\Theta|$.

Example:
For figure 2 the set of index terms is: $\{(f11, f12), f12, f13, (\{f11\}, f13), (f11, \{f13\})\}$
Fig. 2. **Customer Inquiry and Customer Inquiry and Quotation Processing** EPCs.
**Index vector**

*Index vector* for each EPC is determined by assigning a weight to each index term.

*Index term* can be:
- an unmapped function
  - weight = 0
- a mapped function
  - weight representing the similarity with the function to which it is mapped, using one of the similarity functions
- a look-ahead / look-back link
  - weight exponentially decreasing with the number of nodes in the link, using the rationale that links with fewer nodes are more informative than links with more nodes
Index vector

**Definition 15.** Let $P_1$ and $P_2$ be two EPCs with causal footprints $G_1 = (F_1, L_{ib,1}, L_{ia,1})$ and $G_2 = (F_2, L_{ib,2}, L_{ia,2})$, let $M : F_1 \rightarrow F_2$ be a partial injective mapping that associates similar functions, let $\lambda_{|\Theta|}$ be a sequence of index terms as defined in definition 14 and let $\text{Sim}$ be one of the formulae that determines the label similarity of two mapped functions. We define the index vectors, $\overrightarrow{g_1} = (g_{1,1}, g_{1,2}, \ldots, g_{1,|\Theta|})$ and $\overrightarrow{g_2} = (g_{2,1}, g_{2,2}, \ldots, g_{2,|\Theta|})$ for the two EPCs, such that for each index term $\lambda_j$, for $1 \leq j \leq |\Theta|$ and for each $i \in \{1, 2\}$ holds that:

$$g_{i,j} = \begin{cases} 
\text{Sim}(f, f') & \text{if } \exists (f, f') \in M \\
\text{such that } \lambda_j = f \land \lambda_j = f' \\
\frac{\text{Sim}(f, f')}{2^{|\Theta|-1}} & \text{if } \exists (f_s, f) \in L_{ib,i} \\
\text{such that } \lambda_j = (f_s, f) \\
\text{and } (\exists (f, f') \in M \lor \exists (f', f) \in M) \\
\frac{\text{Sim}(f, f')}{2^{|\Theta|-1}} & \text{if } \exists (f, f_s) \in L_{ia,i} \\
\text{such that } \lambda_j = (f, f_s) \\
\text{and } (\exists (f, f') \in M \lor \exists (f', f) \in M) \\
0 & \text{otherwise}
\end{cases}$$

**Example:**
Semantic label similarity. *Index vector* for the rightmost EPC from figure 2 assigns $\text{sem}((f_{11}, f_{12})) \approx 0.69$ to index term $(f_{11}, f_{12})$ and 0 to the other index terms.
Behavioral similarity of the two EPCs, based on their causal footprints = cosine of the angle between their index vectors.

Definition 16. Let $E_1$ and $E_2$ be two EPCs with index vectors $\vec{g}_1$ and $\vec{g}_1$ as defined in definition 15. We define their causal footprint similarity, denoted $\text{simcf}(E_1, E_2)$, as:

$$\text{simcf}(E_1, E_2) = \frac{\vec{g}_1 \times \vec{g}_2}{|\vec{g}_1| \cdot |\vec{g}_2|}$$
Empirical Evaluation Analysis

![Graph showing precision-recall curve]

Fig. 3. Precision-recall curve (precisions are averaged across all 10 queries)
Empirical Evaluation Analysis

Fig. 4. Average precision per search query model
<table>
<thead>
<tr>
<th></th>
<th>Search Engine</th>
<th>Label Similarity</th>
<th>Structural Similarity</th>
<th>Behavioral Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mean Average Precision</strong></td>
<td>0.76</td>
<td>0.80</td>
<td>0.83</td>
<td>0.80</td>
</tr>
<tr>
<td><strong>Pearson Correlation Coefficient</strong></td>
<td>0.03</td>
<td>0.72</td>
<td>0.72</td>
<td>0.73</td>
</tr>
</tbody>
</table>
REFERENCES


Thank you for your attention!