

ESEMÉNYNAPLÓ-LEÍRÓ SZABVÁNYOK ELEMZÉSE

BAKSÁNÉ VARGA ERIKA

A projekt fő célja, hogy neurális háló alapú megoldást találjunk a gyakori folyamatszekvenciák és folyamatgráfok meghatározására az eseménynapló állományok adatai alapján. A folyamatbányász algoritmusok alkalmazása előtt azonban biztosítanunk kell az eseménynaplók megfelelő minőségét és egységes formátumát. Jelen kutatás célja megtalálni azt a formátumot, amelyre a meglévő eseménynaplók egyszerűen konvertálhatók, illeszkedik a neurális hálók inputformátumára, és amely biztosítja a szekvencifeldolgozó algoritmusok hatékony működését.

1. A kutatás célja és lépései

A számítógépen végrehajtott ügyviteli folyamatokat eseménynaplókban tárolják. Ahhoz, hogy az eseménynaplókból ki tudjuk nyerni a leggyakoribb folyamatokat, vagy előre tudjuk jelezni a soron következő folyamatletemet, folyamatbányász eljárásokat kell implementálni. Mivel ezeknek az algoritmusoknak a bemenete eseménynapló, lényeges hogy specifikáljuk ezek formátumát.

Az eseménynapló formátuma az eseményeket rögzítő szoftvertől függ. A Unix-alapú operációs rendszerek például .log kiterjesztésű olvasható állományokban naplózzák az eseményeket, ahol minden sor egy adott esemény bekövetkezésének időpontját és leírását tartalmazza. A Windows operációs rendszernek a 7-es verziótól kezdve saját XML-alapú eseménynapló-leíró formátuma van (EVTX) (korábbi verziókban EVT). Az .evtx kiterjesztésű bináris állományok csak az Event Viewer programmal tekinthetők meg. Mi pedig a kutatás kísérleti fázisában a Task Force on Process Mining oldalról (www.tf-pm.org) letölthető, folyamatbányászathoz előkészített CSV- és XES-állományokkal dolgoztunk.

Érezhető tehát a formátum egységesítésének igénye. Ebben a kutatási jelentésben bemutatom a rendelkezésre álló eseménynapló-leíró szabványokat és megvizsgálom a különböző formátumok közötti konverzió lehetőségét és lépéseit.

2. Kutatási eredmények összesítése

2.1. Elvégzett kísérletek bemutatása

A vizsgált eseménynapló-formátumok: txt, csv, json, evtx, xml, xes, ocel és spmf. Megvizsgáltam ezek között a konverzió lehetőségét azzal a céllal, hogy kijelöljem azt a formátumot, amelyre minden állományt konvertálni fogunk a folyamatbányászathoz megkezdése előtt.

2.1.1. Eseménynaplók konvertálása XES szabványos leírásra

Az XES (eXtensible Event Stream) szabvány egy egységes XML-alapú nyelvet definiál az esemény adatok tárolásához, továbbításához és feldolgozásához. A szabvány két XML Schema leírást tartalmaz: az egyik az XES-eseménynaplók szerkezetét definiálja, a másik egy ilyen naplóhoz tartozó kiterjesztés definícióját határozza meg (<https://xes-standard.org/>).

Az XES-dokumentum szerkezete:

```
<log xes.version="1.0" xes.features="..." xmlns="http://www.xes-standard.org/">
  <trace>
    <event>
      ...
    </event>
  </trace>
</log>
```

Az eseménynapló (log) a végrehajtott folyamatok (trace) tárolására szolgál. Az egyes folyamatokon belül tetszőleges számú esemény (event) fordulhat elő. A log, trace és event jelölők önmagukban nem hordoznak információt, csak az eseménynapló struktúráját definiálják. Az információkat ezek attribútumai tárolják kulcs-érték (key-value) párokban. Hat elemi attribútum létezik, amelyeket az általuk reprezentált adatérték típusa határoz meg (String, Date, Int, Float, Boolean és ID); és két összetett attribútum: a rendezett lista (list) és a halmaz (container). Az összetett attribútumok azonban csak akkor használhatók, ha az „attribútumok egymásba ágyazása” tulajdonság használatát az eseménynapló elején jelezzük a <log> jelölő tagban (<log xes.version="1.0" xes.features="nested-attributes">) és olyan XES-feldolgozót alkalmazunk, amely támogatja ezt a tulajdonságot.

Az XES-szabványban egyik struktúrajelölő elemnek (log, trace, event) sincs előre meghatározott attribútumlistája. Ennek következtében az egyes attribútumok jelentése sincs előre rögzítve, nem egyértelmű. Ezt a hiányosságot küszöbölhetjük ki a kiterjesztések használatával. Az első XES szabvány hét kiterjesztést definiál, amelyek XESEXT XML formátumban vannak megadva: concept, cost, id, lifecycle, organizational, semantic, time. Az újabb, IEEE 1849-2016 szabvány további öt kiterjesztést definiál: artifactlifecycle, micro, swcomm, swevent, swtelemetry.

2.1.1.1. TXT- és CSV-napló átalakítása XES-formátumra

A szöveges naplóállomány tartalma az XES log jelölőelemek közé kerül. Az állomány minden egyes sora külön eseményt (event) ír le, a bekövetkezésük sorrendjében.

Ezeknek az eseményeknek egy adott folyamathoz tartozását vagy nem tárolják, vagy az eseményleírók között szerepel egy folyamatazonosító. Amikor XES-formátumra konvertáljuk a naplót, azokat az eseményeket csoportosítjuk egy trace-be, amelyeknél a folyamatazonosító megegyezik; és ez az adat a trace attribútuma lesz. A folyamatazonosítón kívül is lehetnek olyan eseményleírók, amelyek azonos értékkel rendelkeznek a folyamathoz tartozó összes eseményben. Ezek is a trace attribútumai lesznek az XES-naplóban. Ezeken felül a további eseményleírók az esemény attribútumai.

Amennyiben az események folyamathoz tartozásáról nincs információnk, kétféle feltételezéssel élhetünk. A konkrét naplózási stratégia ismeretében mondhatjuk azt, hogy minden esemény önálló folyamat (ekkor az XES-naplóban minden trace-ben egyetlen event tag lesz); vagy tekinthetünk úgy egy naplóra, hogy az egy adott folyamathoz tartozó eseményeket tárolja (ekkor az XES lognak csak egyetlen trace tagja van és minden event ezen belül kap helyet).

Az XES-naplóban az attribútumokat kulcs-érték párokkal adjuk meg. A kulcsokat a TXT- / CSV-állomány fejlécének (első sorának) kellene tartalmaznia. Amennyiben ez hiányzik, úgy generált kulcsokat alkalmazunk. Az XES-napló az attribútumok típusát is tartalmazza. A legtöbb esetben ez az információ nem nyerhető ki a TXT- / CSV-állományból. Ekkor a legáltalánosabb “szöveg” típust használjuk.

2.1.1.2. JSON-napló átalakítása XES-formátumra

A JSON-(JavaScript Object Notation, JavaScript objektumjelölés) programok közötti strukturált adatcserét lehetővé tevő, szabványos és viszonylag egyszerű szerkezetű formátum. A JSON-fájl szövegfile, tartalma kapcsos zárójel párba zárt vesszővel elválasztott kulcs-érték párosok, amelyek között : van. A kulcs string, és egyedinek kell lennie. Az érték lehet konstans; kapcsos zárójelek közé zárt kulcs-érték párok (asszociatív tömb, amit szokás „objektumnak” is nevezni), vagy szögletes zárójelek közé zárt kulcs-érték párok (rendezett tömb). Ezek az elemek egymásba ágyazhatók: egy objektum elemként tartalmazhat egy másik objektumot vagy tömböt, a tömb elemei is lehetnek objektumok vagy tömbök. Így egészen bonyolult adatszerkezetek is leírhatók.

Egy JSON-naplófájlban egy sorban egy esemény leírása található a megadott formában (JSONLines, vagy JSONL formátum) (<https://jsonlines.org/>). Amikor XES-formátumra alakítjuk, a JSON kulcs-érték párokat leképezzük XES-esemény attribútum-érték párokra. Annak a megállapítása viszont, hogy egy adott esemény melyik folyamathoz tartozik (trace), ugyanúgy történik mint a szöveges naplóállományok esetén. Vagy az eltárolt adatok, vagy a naplózási stratégia alapján döntünk.

2.1.1.3. EVTIX és XML napló átalakítása XES formátumra

Az EVTIX- (<https://www.sciencedirect.com/science/article/pii/S1742287607000424>) napló két lépcsőben alakítható át XES-formátumra. Első lépésben az EvtxParser alkalmazásával XML-formátumra alakítjuk. Az XML-formátumú napló szerkezetét az alábbi ábra mutatja:

```
<Events>
  <Event>
    <System>... </System>
    <EventData>... </EventData>
  </Event>
  <Event>
    ...
  </Event>
</Events>
```

Minden esemény leírása a System elemmel kezdődik. Olyan alapvető adatokat tárol el itt automatikusan a rendszer, mint például az esemény bekövetkezésének helye (a számítógép IP-címe), az esemény időbélyege, vagy az esemény azonosítószáma. Emellett az alábbi elemek közül szerepel még egy az esemény leírásánál: BinaryEventData, DebugData, EventData, ProcessingErrorData, Rendering-Info, vagy UserData. Leggyakrabban az EventData elemmel találkozhatunk, ami az eseményt kiváltó alkalmazás által átadott paramétereket tartalmazza.

Látható, hogy ez a formátum is az események szekvenciáját tartalmazza, ugyanúgy mint a szöveges leíró állományok. Az események XES-folyamatokhoz (trace) rendelését az eltárolt adatok, vagy a naplózási stratégia alapján határozhatjuk meg. Viszont az XML event tag egy az egyben megfeleltethető az XES event tagnak.

2.1.2. XES-formátum átalakítása OCEL szabványos leírásra

Látható, hogy az XES folyamatcentrikus leírásmód, azaz központi eleme a trace, és az esemenynaplóban rögzített minden tevékenységet egy folyamathoz kötünk. Az információs rendszerekben azonban, a párhuzamosan zajló folyamatok során egy adott esemény rendszerint több, különböző folyamatban érintett objektumra vonatkozik. Azaz az XES-szabvány szerinti folyamatcentrikus modellezéskor az adott eseményt egyidejűleg csak egy objektumtípus szemszögéből vizsgáljuk. Ezzel viszont információt veszünk. Ezért dolgozták ki az OCEL (Object-Centric Event Logs) objektumcentrikus esemenynapló-formátumot (<http://www.ocel-standard.org/>).

Az OCEL az objektumcentrikus eseménynaplók szabványos leíró nyelve. Szintaktikáját tekintve nagyon hasonlít az XES-szabványhoz. A legfontosabb különbségek a két nyelv között:

- Az OCEL nyelv nem használja a „trace” (folyamatpéldány) jelölőelemet.
- Az OCEL bevezeti az objektum („object”) jelölőelemet.

Az OCEL-naplóállomány megadható XML- és JSON-formátumban is. Az elején globális listában definiáljuk a naplóban szereplő objektumtípusokat és az eseményeket leíró attribútumokat. Ezután következik a bekövetkezett események leírása, ami az eseményattribútumok értékét és az érintett objektumok referenciáját tartalmazza.

Az XES-formátum OCEL-leírásra konvertálásakor a trace jellemzői az események attribútumai között fognak szerepelni. Az XES-eseményattribútumokból osztályokat képzünk. Ezekből a típusokból lesznek az objektumok, amelyeket külön definiálunk és az adott esemény csak a referenciájukat tárolja. Azok az XES-eseményattribútumok, amelyek közvetlenül az eseményhez kapcsolódnak (nem pedig az esemény által érintett valamely objektumhoz), továbbra is eseményattribútumok lesznek.

2.1.3. Szöveges napló átalakítása SPMF-formátumra

Az SPMF (Sequence Pattern Mining Format) (<http://www.philippe-fournier-viger.com/spmf/>) egy Java könyvtár, ami elemhalmazok és szekvenciák bányászatát támogató algoritmusok implementációját foglalja magában. Az input adatállományokhoz saját formátumot (és erre a formátumra konvertáló eljárásokat) biztosít. Az SPMF formátumban minden sor egy esemény leírása számsorozattal. A számok az eseményhez tartozó attribútum-érték párok sorszámai. Az attribútum-érték párok metaadatok, azaz az adatbányász algoritmusok nem használják fel ezeket, csak a kapott eredmények értelmezhetősége miatt tartoznak hozzá a formátumleíráshoz.

A szöveges naplóállomány átalakítása során egy SPMF-szekvencia-adatbázist hozunk létre. Ez tartalmazza az attribútum-érték párok definícióját és az eseményeket leíró szekvenciákat. Az attribútum-érték párok definíciós sora @ITEM = sorszám = érték alakú, ahol a sorszám 1-től kezdődően folytonosan növekszik és minden egyedi értékre különböző. Egy esemény adatainak definícióját követően felírjuk az eseményt, mint az egymás után következő értékek szekvenciáját, ahol az egyes értékek között –1 az elválasztó karakter, és a szekvencia végét –2 zárja. Ennek a formátumnak az előnye az információtárolás minimalizálása és ezáltal az adatbányász algoritmusok hatékonyságának a biztosítása.

2.2. Kiértékeléshez használt referenciafolyamatok és értékek bemutatása

Az ügyviteli folyamatokkal kapcsolatban eltárolt adatok:

case id	customer request			
id	id	description	priority	source

activity						agent			
id	time	state	type	category	input	id	group	role	host

Adatalem	Leírás
Case id	az ügyviteli folyamat egyedi azonosítója (egy folyamat több tevékenységet foglalhat magában)
Request id	a beérkezett kérés azonosítója (előfordulhat, hogy egy kérés kiszolgálása több ügyviteli folyamatot is elindít)
Request description	a kérés szöveges leírása
Request priority	a kérés prioritása, sürgőssége (előre definiált prioritási szintek alapján)
Request source	a kérés forrása (az ügyfél azonosítását szolgáló adatok)
Activity id	az ügyviteli folyamat során elvégzett tevékenység azonosítója (előre definiált tevékenységlista alapján)
Activity time	a tevékenység végrehajtási ideje (rendszer időbélyeg)
Activity state	a tevékenység állapota (pl.: befejezett, felfüggesztett, ütemezett stb.) (előre definiált állapotlista alapján)
Activity type / category	előre definiált tevékenység-hierarchia alapján a tevékenység besorolása (típusa, kategóriája)
Activity input	a tevékenység során rögzített adatérték (ha volt)
Agent id	a tevékenységet végrehajtó ügyintéző azonosítója
Agent group	a tevékenységet végrehajtó ügyintéző melyik szervezeti egységhez tartozik
Agent role	a tevékenységet végrehajtó ügyintéző munkaköre
Agent host	a tevékenység végrehajtásának helye (a számítógép azonosítója)

A különböző formátumok közötti konverziós eljárások kidolgozása során felhasznált naplóállományok, amelyek egy példa ügyviteli folyamat 4 eseményének adatait tartalmazzák:

TXT, CSV:

```
Case-id;Request-id;Request-desc;Request-priority;Request-source;Activity-id;Activity-time;Activity-state;Activity-type;Activity-category;Activity-input;Agent-id;Agent-group;Agent-role;Agent-host
111;222;invoice complaint;urgent;333;441;2021-11-07T02:26:35.000+01:00;completed;register request;invoicing;555;customer service;clerk;193.4.5.6
111;222;invoice complaint;urgent;333;442;2021-11-08T02:26:35.000+01:00;completed;examine request;invoicing;invoice data;666;accounting;clerk;194.6.5.8
111;222;invoice complaint;urgent;333;443;2021-11-10T02:26:35.000+01:00;completed;decide;invoicing;decision;777;accounting;administrator;195.4.5.6
111;222;invoice complaint;urgent;333;444;2021-11-11T02:26:35.000+01:00;completed;reject request;invoicing;555;customer service;clerk;193.4.5.6
```

JSONL:

```
{"Activity-category":"invoicing","Activity-id":441,"Activity-input":"","Activity-state":"completed","Activity-time":"2021-11-07T02:26:35.000+01:00","Activity-type":"register request","Agent-group":"customer service","Agent-host":"193.4.5.6","Agent-id":555,"Agent-role":"clerk","Case-id":111,"Request-desc":"invoice complaint","Request-id":222,"Request-priority":"urgent","Request-source":333}
{"Activity-category":"invoicing","Activity-id":442,"Activity-input":"invoice data","Activity-state":"completed","Activity-time":"2021-11-08T02:26:35.000+01:00","Activity-type":"examine request","Agent-group":"accounting","Agent-host":"194.6.5.8","Agent-id":666,"Agent-role":"clerk","Case-id":111,"Request-desc":"invoice complaint","Request-id":222,"Request-priority":"urgent","Request-source":333}
{"Activity-category":"invoicing","Activity-id":443,"Activity-input":"decision","Activity-state":"completed","Activity-time":"2021-11-10T02:26:35.000+01:00","Activity-type":"decide","Agent-group":"accounting","Agent-host":"195.4.5.6","Agent-id":777,"Agent-role":"administrator","Case-id":111,"Request-desc":"invoice complaint","Request-id":222,"Request-priority":"urgent","Request-source":333}
{"Activity-category":"invoicing","Activity-id":444,"Activity-input":"","Activity-state":"completed","Activity-time":"2021-11-11T02:26:35.000+01:00","Activity-type":"reject request","Agent-group":"customer service","Agent-host":"193.4.5.6","Agent-id":555,"Agent-role":"clerk","Case-id":111,"Request-desc":"invoice complaint","Request-id":222,"Request-priority":"urgent","Request-source":333}
```

XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <row>
    <Case-id>111</Case-id>
    <Request-id>222</Request-id>
    <Request-desc>invoice complaint</Request-desc>
    <Request-priority>urgent</Request-priority>
    <Request-source>333</Request-source>
    <Activity-id>441</Activity-id>
    <Activity-time>2021-11-07T02:26:35.000+01:00</Activity-time>
    <Activity-state>completed</Activity-state>
```

```
<Activity-type>register request</Activity-type>
<Activity-category>invoicing</Activity-category>
<Activity-input></Activity-input>
<Agent-id>555</Agent-id>
<Agent-group>customer service</Agent-group>
<Agent-role>clerk</Agent-role>
<Agent-host>193.4.5.6</Agent-host>
</row>
<row>
<Case-id>111</Case-id>
<Request-id>222</Request-id>
<Request-desc>invoice complaint</Request-desc>
<Request-priority>urgent</Request-priority>
<Request-source>333</Request-source>
<Activity-id>442</Activity-id>
<Activity-time>2021-11-08T02:26:35.000+01:00</Activity-time>
<Activity-state>completed</Activity-state>
<Activity-type>examine request</Activity-type>
<Activity-category>invoicing</Activity-category>
<Activity-input>invoice data</Activity-input>
<Agent-id>666</Agent-id>
<Agent-group>accounting</Agent-group>
<Agent-role>clerk</Agent-role>
<Agent-host>194.6.5.8</Agent-host>
</row>
<row>
<Case-id>111</Case-id>
<Request-id>222</Request-id>
<Request-desc>invoice complaint</Request-desc>
<Request-priority>urgent</Request-priority>
<Request-source>333</Request-source>
<Activity-id>443</Activity-id>
<Activity-time>2021-11-10T02:26:35.000+01:00</Activity-time>
<Activity-state>completed</Activity-state>
<Activity-type>decide</Activity-type>
<Activity-category>invoicing</Activity-category>
<Activity-input>decision</Activity-input>
<Agent-id>777</Agent-id>
<Agent-group>accounting</Agent-group>
<Agent-role>administrator</Agent-role>
<Agent-host>195.4.5.6</Agent-host>
</row>
<row>
```



```

<Case-id>111</Case-id>
<Request-id>222</Request-id>
<Request-desc>invoice complaint</Request-desc>
<Request-priority>urgent</Request-priority>
<Request-source>333</Request-source>
<Activity-id>444</Activity-id>
<Activity-time>2021-11-11T02:26:35.000+01:00</Activity-time>
<Activity-state>completed</Activity-state>
<Activity-type>reject request</Activity-type>
<Activity-category>invoicing</Activity-category>
<Activity-input></Activity-input>
<Agent-id>555</Agent-id>
<Agent-group>customer service</Agent-group>
<Agent-role>clerk</Agent-role>
<Agent-host>193.4.5.6</Agent-host>
</row>
</root>

```

2.3. Kiértékelések eredményeinek bemutatása

A következő táblázatok a formátumátalakítási lépéseket foglalják össze.

TXT, CSV, JSONL, XML → XES

A vastagon szedett XES-attribútumkulcsok szabványos kiterjesztések, amelyek alkalmazása segíti az egységes szemantikai értelmezést. Az azonos színnel megjelölt attribútumokból pedig összetett attribútumok alkothatók, ami támogatja az objektumközpontú ábrázolást.

Forrás	XES bennfoglaló tag	XES-attribútum
Case id	<trace>	<string key="concept:name" value="111"/>
Request id	<trace>	<string key="request-id" value="222">
Request description	<trace>	<string key="request-desc" value="invoice complaint">
Request priority	<trace>	<string key="request-priority" value="urgent">
Request source	<trace>	<string key="request-source" value="333">
Activity id	<event>	<string key="concept:name" value="441"/>
Activity time	<event>	<date key="time:timestamp" value="2009-11-25T12:45:32.345+02:00" />
Activity state	<event>	<string key="lifecycle:transition" value="completed"/>

Forrás	XES bennfoglaló tag	XES-attribútum
Activity type	<event>	<string key="activity-type" value="register request" />
Activity category	<event>	<string key="activity-category" value="invoicing" />
Activity input	<event>	<string key="activity-input" value="" />

Forrás	XES bennfoglaló tag	XES-attribútum
Agent id	<event>	<string key="org:resource" value="555" />
Agent group	<event>	<string key="org:group" value="customer service" />
Agent role	<event>	<string key="org:role" value="clerk" />
Agent host	<event>	<string key="agent-host" value="193.4.5.6" />

XES → OCEL

OCEL-ben egy esemény leírása az eseményattribútumok értékét és az érintett objektumok referenciáját tartalmazza. Az ugyanahhoz a folyamathoz tartozó eseményeket az <events> ... </events> jelölő elemek között adjuk meg. Az XES-folyamatazonosítót bele kell kódolni az események azonosítójába.

Forrás	OCEL-esemény	OCEL-leírás
Case id + Activity id	<string key="name" value="id"/>	<string key="id" value="111-441"/>
Event time	<string key="name" value="time"/>	<date key="timestamp" value="2009-11-25T12:45:32.345+02:00" />
Event state	<string key="name" value="state"/>	<string key="state" value="completed"/>

Forrás	OCEL-objektum	OCEL-leírás
Request id	<string key="type" value="request"/>	<string key="id" value="222"/>
Request description	<string key="type" value="request"/>	<string key="type" value="invoice complaint"/>
Request priority	<string key="type" value="request"/>	<string key="priority" value="urgent" />
Request source	<string key="type" value="request"/>	<string key="source" value="333" />
Activity id	<string key="type" value="activity"/>	<string key="id" value="222"/>

Forrás	OCEL-objektum	OCEL-leírás
Activity type	<string key="type" value="activity"/>	<string key="type" value="register request" />
Activity category	<string key="type" value="activity"/>	<string key="category" value="invoicing" />
Activity time	<string key="type" value="activity"/>	<date key="timestamp" value="2009-11-25T12:45:32.345+02:00" />
Activity state	<string key="type" value="activity"/>	<string key="state" value="completed"/>
Activity input	<string key="type" value="activity"/>	<string key="input" value="" />
Agent id	<string key="type" value="agent"/>	<string key="id" value="555"/>
Agent role	<string key="type" value="agent"/>	<string key="type" value="clerk" />
Agent group	<string key="type" value="agent"/>	<string key="group" value="customer service" />
Agent host	<string key="type" value="agent"/>	<string key="host" value="193.4.5.6"/>

TXT, CSV → SPMF

Az SPMF-szekvencia-adatbázisban metaadat-definíciók és szekvenciák követik egymást. Az üresen hagyott mezők figyelembevételéhez, illetve azért, hogy minden szekvencia azonos hosszúságú legyen, bevezettem a helykitöltő „null” értéket.

@ITEM=1=111

@ITEM=2=222

@ITEM=3=invoice complaint

@ITEM=4=urgent

@ITEM=5=333

@ITEM=6=441

@ITEM=7=2021-11-07T02:26:35.000+01:00

@ITEM=8=completed

@ITEM=9=register request

@ITEM=10=invoicing

@ITEM=11=null

@ITEM=12=555

@ITEM=13=customer service

@ITEM=14=clerk

@ITEM=15=193.4.5.6

1 -1 2 -1 3 -1 4 -1 5 -1 6 -1 7 -1 8 -1 9 -1 10 -1 11 -1 12 -1 13 -1 14 -1 15 -1 -2

2.4. Eredményeket szemléltető képernyőképek

A minta szöveges naplóállomány első sorának XES-szabvány szerinti leírása:

```
<?xml version="1.0" encoding="UTF-8" ?>
<log xes.version="1.0" xes.features="nested-attributes" openxes.version="1.0RC7">
  <extension name="Time" prefix="time" uri="http://www.xes-standard.org/time.xesext"/>
  <extension name="Lifecycle" prefix="lifecycle" uri="http://www.xes-standard.org/lifecycle.xesext"/>
  <extension name="Concept" prefix="concept" uri="http://www.xes-standard.org/concept.xesext"/>
  <extension name="Organizational" prefix="org" uri="http://code.fluxicon.com/xes/org.xesext" />
  <string key="concept:name" value="XES Event Log"/>
  <trace>
    <string key="concept:name" value="111"/>
    <string key="request" value="222">
      <string key="description" value="invoice complaint" />
      <string key="priority" value="urgent" />
      <string key="source" value="333" />
    </string>
    <event>
      <string key="concept:name" value="441"/>
      <date key="time:timestamp" value="2009-11-25T12:45:32.345+02:00" />
      <string key="lifecycle:transition" value="completed"/>
      <string key="type" value="register request" />
      <string key="category" value="invoicing" />
      <string key="input" value="" />
      <string key="agent" value="555">
        <string key="org:group" value="customer service" />
        <string key="org:role" value="clerk" />
        <string key="host" value="193.4.5.6" />
      </string>
    </event>
  </trace>
</log>
```

A minta szöveges naplóállomány első sorának OCEL-szabvány szerinti leírása:

```
<?xml version='1.0' encoding='UTF-8'?>
<log>
  <global scope="event">
    <string key="id" value="__INVALID__"/>
    <string key="timestamp" value="__INVALID__">
    <string key="state" value="__INVALID__"/>
    <string key="omap" value="__INVALID__"/>
  </global>
  <global scope="object">
    <string key="id" value="__INVALID__"/>
    <string key="type" value="__INVALID__"/>
  </global>
  <global scope="log">
    <string key="version" value="0.1" />
    <string key="ordering" value="timestamp" />
    <list key="attribute-names">
      <string key="name" value="request-priority"/>
      <string key="name" value="request-source"/>
      <string key="name" value="activity-category"/>
    </list>
  </global>
</log>
```

```

<string key="name" value="activity-time"/>
<string key="name" value="activity-state"/>
<string key="name" value="activity-input"/>
<string key="name" value="agent-group"/>
<string key="name" value="agent-host"/>
</list>
<list key="object-types">
  <string key="type" value="request"/>
  <string key="type" value="activity"/>
  <string key="type" value="agent"/>
</list>
</global>
<events>
  <event>
    <string key="id" value="111-441"/>
    <date key="timestamp" value="2020-07-09T08:20:01.527+01:00"/>
    <string key="state" value="completed"/>
    <list key="omap">
      <string key="object-id" value="222"/>
      <string key="object-id" value="441"/>
      <string key="object-id" value="555"/>
    </list>
  </event>
  ...
</events>
<objects>
  <object>
    <string key="id" value="222"/>
    <string key="type" value="invoice complaint"/>
    <list key="ovmap">
      <string key="request-source" value="333" />
      <string key="request-priority" value="urgent" />
    </list>
  </object>
  <object>
    <string key="id" value="441"/>
    <string key="type" value="register request"/>
    <list key="ovmap">
      <string key="activity-category" value="invoicing" />
      <date key="activity-time" value="2020-07-09T08:20:01.527+01:00"/>
      <string key="activity-state" value="completed"/>
      <string key="activity-input" value="" />
    </list>
  </object>
  <object>
    <string key="id" value="555"/>
    <string key="type" value="clerk"/>
    <list key="ovmap">
      <string key="agent-group" value="customer service" />
      <string key="agent-host" value="193.4.5.6" />
    </list>
  </object>
</objects>
</log>

```

A teljes minta szöveges naplóállomány SPMF-formátum szerinti szekvencia leírása:

1 -1 2 -1 3 -1 4 -1 5 -1 6 -1 7 -1 8 -1 9 -1 10 -1 11 -1 12 -1 13 -1 14 -1 15 -1 -2
 1 -1 2 -1 3 -1 4 -1 5 -1 16 -1 17 -1 8 -1 18 -1 10 -1 19 -1 20 -1 21 -1 14 -1 22 -1 -2
 1 -1 2 -1 3 -1 4 -1 5 -1 23 -1 24 -1 8 -1 25 -1 10 -1 26 -1 27 -1 21 -1 28 -1 29 -1 -2
 1 -1 2 -1 3 -1 4 -1 5 -1 30 -1 31 -1 8 -1 32 -1 10 -1 11 -1 12 -1 13 -1 14 -1 15 -1 -2

2.5. Eredményeket szemléltető diagramok

Az XES-, az OCEL- és az SPMF-formátumok szerint leírt eseménynaplókat a napló terjedelme (L) alapján hasonlítottam össze. A napló terjedelme az állomány hossza (a kódsorok száma), amelyet a tárolt információelemek számának függvényeként határoztam meg. Ez a mérőszám azért lényeges, mert befolyásolja a naplófeldolgozó eljárások futási sebességét.

L – terjedelem

FIX – a dokumentum információelemektől független sorainak száma

T – folyamatok száma (trace)

TA – folyamat attribútumainak száma

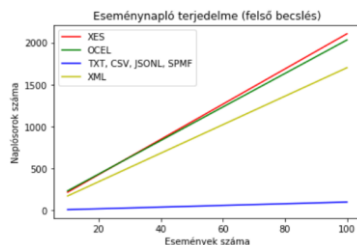
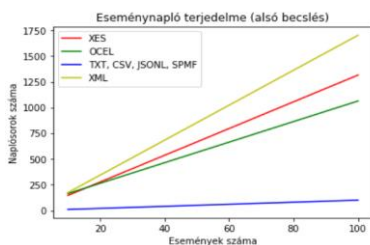
E – események száma

EA – események attribútumainak száma

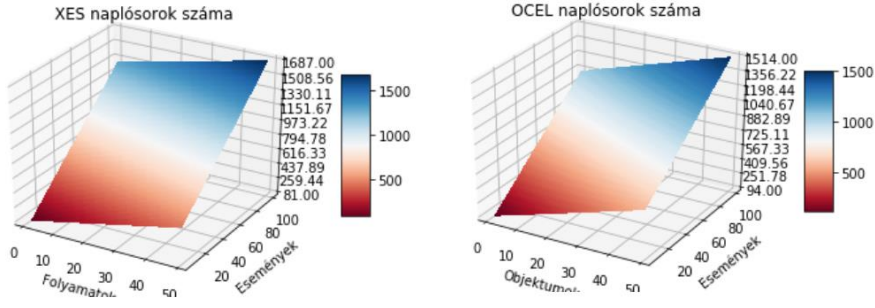
O – objektumok száma Opreambulom hossza PRE

Dokumentum	Terjedelem	ERPA alsó becslés	ERPA felső becslés
XES	$FIX + (T*TA) + (E*EA)$	$T = 1$ $8 + 8 + E*13$	$T = E$ $8 + E*8 + E*13$
OCEL	$FIX + EA + (E*10) + (O*EA)$	$O = 3$ $34 + E*10 + 30$	$O = E$ $34 + E*10*2$
TXT, CSV, JSONL, SPMF	E	E	E
XML	$FIX + E * (TA + EA + 2)$	$3 + E*17$	$3 + E*17$

A terjedelem növekedésének szemléltetése csak az események számának függvényében:



A terjedelem növekedése mindkét változó figyelembevétele mellett:



Az ábrából látható, hogy az objektumok számának növekedése nagyobb mértékben hozzájárul a terjedelem növekedéséhez, mint a folyamatok számának növekedése. Az események leírása viszont hosszabb a folyamatcentrikus megközelítésben, mint az objektumközpontúban. Ezért amikor az eseményszám nagy, az XES-dokumentum hosszabb, mint az OCEL.

3. Összegzés

A feladatom az volt, hogy összehasonlítsam a különböző eseménynapló-formátumokat abból a szempontból, hogy melyik a legalkalmasabb a folyamatbányász algoritmusokhoz, azaz melyik biztosítja az algoritmusok leghatékonyabb működését.

A szöveges formátumok (txt, csv, xml, json) nem szabványosak, ezért egységesítésük nehézkes, egyedi eljárásokat igényel. Az XES és OCEL szabványos formátumok, egyszerűbb az automatikus feldolgozásuk, de használatuk rendkívül terjedelmes naplóállományt eredményez.

A 2.5. szakaszban bemutatott számítások és grafikonok mutatják, hogy a leg-tömörebb leírás a szöveges, ahol egy eseményt egy sorban adunk meg. Ezek közül az SPMF-formátum az, amelyik kifejezetten folyamatbányászathoz van előké-szítve, ezért a továbbiakban ezt a formátumot preferáljuk.