KORRUPCIÓKUTATÓ KÖZPONT BUDAPEST

Közbeszerzési eljárások dokumentumainak adatfeldolgozása 1998-2004

Módszertani elemzés

Public Procurement 4 You

8. Riport

Budapest, 2014. március

A riportot a Korrupciókutató-központ Budapest (CRCB) kutatócsoportja készítette. A kutatás folytatását a központ pénzügyi erőforrásai, és a résztvevők önkéntes munkája tette lehetővé.

A kutatás folytatásához minden anyagi segítséget szívesen fogadunk, illetve várjuk önkéntesek jelentkezését (info@crcb.eu).

Közbeszerzési eljárások dokumentumainak adatfeldolgozása 1998-2004

/ Data extraction from public procurement documents 1998-2004

Nyolcadik riport / 8th Report

Együttműködő partnerek:

Erőforrástérkép projekt (MTA KRTK KTI): http://regionaldata.org
Átlátható Állam: http://www.atlathatoallam.hu/
3gteam: http://www.3gteam.hu/

A riportot írta: Fóra Gyula

Munkatársak:

Czibik Ágnes	közgazdász
Fazekas Mihály	Ph.D. hallgató, University of Cambridge
Fóra Gyula	egyetemi hallgató
Orbán Júlia	közgazdász
Tóth Bence	közgazdász
Tóth István János	Ph.D, tudományos főmunkatárs, MTA KRTK KTI

Külső szakértők:

Dr. Kelemen Zoltán	ügyvéd
Gyenese Jenő	programtervező matematikus
Nagy Zoltán	közgazdász
Siposs Zoltán	újságíró
Uhrin Tamás	programtervező matematikus
Dr. Volford Anita Adrienn	ügyvédjelölt

Corruption Research Center Budapest
e-mail: info@crcb.eu
internet: http://www.crcb.eu/

A kézirat lezárva: 2014. március 28.

Tartalomjegyzék

Abstract

Although the Hungarian Public Procurement Authority has made all the information publicly available online for the public procurement tenders between 1998 and 2004, the data format is inappropriate for statistical analysis. The information is stored in basic HTML files which does not provide any interface for sorting and searching among the data. In this technical paper we will describe our data extraction process which we used to turn the HTML based information into database format by extracting relevant fields of information. The Python programming language was used for data cleaning and extraction, which resulted in a database suitable for further analysis. At the end of the paper, basic statistics about the dataset is shown to provide some examples for the usability of this dataset.

Bevezetés

A Közbeszerzési Hatóság által nyilvánosságra bocsátott 1998 és 2004 közötti közbeszerzési adatok eddig csak elemzési célra alkalmatlan formában voltak elérhetőek. A HTML formátumban tárolt pályázati adatok nem biztosítottak semmilyen kézenfekvő lehetőséget az adatok rendszerezésére, keresésére és akármilyen jellegű statisztikai vizsgálatára. A következőkben bemutatott adatfeldolgozó eljárás célja, hogy az elérhető HTML dokumentumokból a lehető legtöbb releváns információt kinyerjük és egy rendszerezett adatbázisba rögzítsük, ami utána lehetővé teszi a további elemző munkát.

Fejlesztési környezet

Az adatok letöltéséhez és az adatkinyeréshez is Python programnyelv 2.7-es verzióját használtam.

Felhasznált nem standard könyvtárak

- urrlib: weboldalak letöltése
- nltk: Natural Language Processing Toolkit, alapvető nyelvfeldolgozási funkciók pl. szótövek levágása, valamint HTML szöveggé alakítás
- numpy: Gyors tömbműveletek
- pandas: DataFrame adatstruktúra (Fejlécezett tömb) sok hasznos feldolgozási funkcióval, pl. mentés Excel formátumba, adatelemző funkciók
- sklearn: Szöveg kategorizáláshoz adatbányászati eszközök, valamint függvények python objektumok lemezre mentéséhez(joblib)
- unidecode: Ékezetes szövegek ékezet mentessé alakítása

Közbeszerzési adatok letöltése

A régi közbeszerzési honlapon (http://regi.kozbeszerzes.hu/static/KEarchiv/index.html) elérhető adatokat mappákba rendezve töltöttem le. A HTML fájlok letöltését a download_data(folder,year) függvény végzi, ami bejárja az évhez tartozó linkeket és letölti a megfelelő dokumentumokat mappákba rendezve. Az adatokat HTML formátumban mentettem el a sorszámok alapján. A 2004-es adatok letöltésére külön a download_data2004() függvény használható az eltérő weblap struktúra miatt.

Előkészítő műveletek

Mielőtt lehetséges lenne, a változók kinyerése, fel kell darabolni a szöveges fájlokat mezők szerint. Ezt egyértelműen megtehetjük az eredményhirdetések alapvető formai tulajdonságai alapján.

Ezt a műveletet az extract_fields(folder, fout, year) függvény végzi, ami egy adott mappában található eredményhirdetéseket mezőkre bontva eltárolja. A függvény felülről lefelé halad a fájlokban és soronként keresi az előre megadott kulcsszavakat (vagy mondatrészeket) amik egyértelműen beazonosítják az eredményhirdetés különböző részeit. Ezek alapján feldarabolja a szöveges fájl a mezőknek megfelelő darabokra, amit ez után arra használhatunk, hogy kinyerjük a megfelelő információt.

A mezőket egy Python szótárban tároljuk ahol a kulcsok a megfelelő mezők nevei és az elemek a hozzá tartotó szövegdarabok. Ezeket a Python objektumokat is külön elmentettem ezzel is szétbontva a feldolgozási műveleteket.

Pl: {’ajanlsz’: ’tizenegy’, ’nyertesn’: ’ Renault Trucks Hungária Kft., 2046 Törökbálint, Tó Park.’}

Információk kinyerése

A mezők kinyeréséhez ezután végigmegyünk az egyes eredményhirdetésekhez készített szótárakon és a megfelelő kulcs alatt található szövegekre alkalmazzuk a feldolgozó függvényeket.

A szöveges mezők feldolgozására használt legfontosabb függvények:

- **extract_writtennums(field)**
  Egy mezőben található szöveggel kiírt számot, rendes számmá alakít, elsősorban az ajánlatok számának kinyeréséhez. A szöveget feldarabolja, majd a szavakat egyenként megvizsgálja, hogy számok-e (read_num()).
- **to_datetime(field)**
  Szöveges magyar dátumokat alakít a Python által kezelhető dátum formátummá. Megkeresi az első dátumot a megadott karakterláncban Regular Expression segítségével, majd ezt a megfelelő formátumra alakítja.
- **extract_address(field)**
  Egy mezőben található címeket keresi meg Regular Expression (előre meghatározott szövegmintázatok)-ök segítségével, majd a címeket (isz, város, utca házszám) bontásban visszaadja. Elsősorban a településre és irányítószámra pontos az utcanevek sok esetben hiányoznak. A nem budapesti irányítószám nélküli címeket nem tudja kezelni.
- **extract_nevekcimek(field)**
  A mezőt sorokra bontva megkeresi azokat a sorokat, amelyek egy cég nevét és címét tartalmazzák. A cégnevek beazonosításához először címeket keres mivel ezek sokkal könnyebben beazonosíthatók és mindig ott vannak a cégnév mellett vagy alatt. A nyerteseket és ajánlattevőket tartalmazó mezőben ilyen formában vannak tárolva az információk.
- **extract_nums(field)**
  A mezőből Regular Expression-ök segítségével megtalálja, majd összeadja a forint összegeket. Pl.: 500 ezer FT, 20 300 ft, 3 m ft stb.
  Külön megnézi a függvény, hogy valószínűleg db-áras termékről van e szó, ezt a leírásból próbáltam kikövetkeztetni. Darab-ár esetén -1-et ad vissza értéknek.
- **extract_targy(field)**
  A mezőben található szöveget egy kategorizáló eljárás segítségével az előre megadott kategóriák valamelyikébe sorolja. A feladat nagyon nehéz és automatikusan nem nagyon lehet 70% pontosság fölé kerülni, ezért mindenképpen kézzel kell átnézni, ha erre az információra szükség van.
- **extract_eredmeny(field)**
  Az eredményességet mutató szavakat keresi a mezőben, ezek alapján három kategória egyikébe sorolja: eredményes, eredménytelen, részben eredményes
- **extract_eljaras(field)**
  A különböző eljárástípusokat mutató szavakat keresi a mezőben. (Pl. Nyílt, Előminősítési, Zárt, stb.)
- **extract_aktip(field)**
  A különböző vállalati formákat nyeri ki a megadott vállalat nevéből.(Pl. Önkormányzat, részvénytársaság stb.) Igény szerint bővíthető a lista.

Az imént felsorolt függvények segítségével miután kinyertük a szükséges információkat, az adatokat egy DataFrame objektumban helyezzük el. Ez gyakorlatilag egy fejlécezett táblázatnak feleltethető meg, aminek sorai az egyes rekordok. A statisztikai elemzés során érdemes lehet ezt az objektumot használni a remek kezelhetősége miatt.

A létrehozott DataFrame objektumokat is külön a lemezre mentettem, ez gyakorlatilag megfelel az Excel táblázatoknak (azok ebből lettek generálva) de már egy Pythonban kezelhető formában, ami összetettebb elemzési célokra alkalmasabb lehet.

Kinyert változók

Mezőnév	Felhasznált függvények	Megjegyzés	Státusz
AKN (Ajánlatkérő neve)		A megfelelő mező első vagy első két sora. Előfordulhat, hogy több mint egy ajánlatkérő volt, de ez nagyon kevés esetet érint és nehezen kezelhető. Ebben az esetben az elsőt láthatjuk.	95%
AKTIP	extract_aktip()	Megfelelően működik	100%
AKC (T,IRSZ,UT) (Ajánlatkérő cím részei)	extract_address()	Nem-konvencionális címzésnél probléma lehet (adatok nagyon kis részét érinti).	90%
AJSZ (Ajánlatok száma)	extract_writtennums()	Megfelelően működik	100%
ELJ_TIPUS (Eljárás típusa)	extract_eljaras()	Megfelelően működik	100%
ELB_SZ (Elbírálás szempontjai)		Ez a mező a nyers szöveget tartalmazza	100%
ATN (Ajánlattevők nevei)	extract_nevekcimek()	Nagyon ritkán eltérő formátum miatt problémák, csak manuális kezelhető + nem-konvencionális címzés	95%
AT_TIP (Ajánlattevők típusa)	extract_aktip()	Megfelelően működik	100%
ATC(T,IRSZ,UT) (Ajánlattevők címe)	extract_nevekcimek()	Nagyon ritkán eltérő formátum miatt problémák, csak manuális kezelhető + nem-konvencionális címzés	95%
AT_SZ (Ajánlattevők száma)	A nevekből triviális	Megfelelően működik.	100%
NYN (Nyertes neve)	extract_nevekcimek()	Nagyon ritkán eltérő formátum miatt problémák, csak manuális kezelhető + nem-konvencionális címzés	95%
NY_TIP (Nyertes típusa)	extract_aktip()	Megfelelően működik.	95%
NYC (T,IRSZ,UT)	extract_nevekcimek()	Nagyon ritkán eltérő formátum miatt problémák, csak manuális kezelhető + nem-konvencionális címzés	95%
NY_SZ (Nyertesek száma)	A nyertesek neveiből triviális	Megfelelően működik.	100%
URL			100%

TARGY (Beszerzés tárgya)	extract_targy()	A kategorizálás pontosságán múlik. Csak manuálisan kezelhető.	60%
T_raw		A tárgy teljes szövege	100%
OSSZEG (Beszerzés értéke)	extract_nums()	Óra/napi/éves esetén -1, sok hiba. Nem egyértelmű sok helyen, manuálisan kezelhető.	85%
EREDMENYES (Eljárás eredményessége)	extract_eredmeny()	Megfelelően működik	100%
EH_D (EV,HO,NAP) (Eredményhirdetés napja)	to_datetime()		100%
FELAD_D(EV,HO,NAP) (Ajánlat feladása)	to_datetime()	Megfelelően működik	100%
KVETEL_D(EV,HO,NAP) (Ajánlat kézhezvétele)	to_datetime()	Megfelelően működik	100%
KOZZT_D(EV,HO,NAP) (Közzététel)	to_datetime()	Megfelelően működik	100%
ELB_D(EV,HO,NAP) (Elbírálás)	to_datetime()	Megfelelően működik	100%
ALV (Alvállalkozó adatok)		A mező a nyers szöveget tartalmazza	
YEAR (Az eljárás éve)			100%

Kategorikus változók leírása

AKTIP

'alapitvany': 1
'birosag': 2
'bt': 3
'egyeb': 4
'egyetem': 5
'kft': 6
'kht': 7
'konzorcium': 8
'korhaz': 9
'miniszterium': 10
'onkormanyzat': 11
'rt': 12
'vallalkozo': 13

NY_TIP, AT_TIP

'alapitvany': 1
'bt': 2
'egyeb': 3
'egyetem': 4
'kft': 5
'kht': 6
'konzorcium': 7
'korhaz': 8
'miniszterium': 9
'onkormanyzat': 10
'rt': 11
'vallalkozo': 12

ELJ_TIP

'gyorsított tárgyalásos': 1
'közzététel n tárgyalásos': 2
'k.n.t előmin': 3
'közzétét. tárgyalásos': 4
'nyílt': 5
'nyílt előmin.': 6
'tárgyalásos': 7
'tárgyalásos előmin: 8

TARGY

'it beszerzes': 1
'epites': 2
'etkeztetes': 3
'gepbeszerzes': 4
'orvosi beszerzes': 5
'szolgaltatas': 6
'targybeszerzes': 7
'tisztitas': 8

Alapinformációk a létrehozott adatbázisról

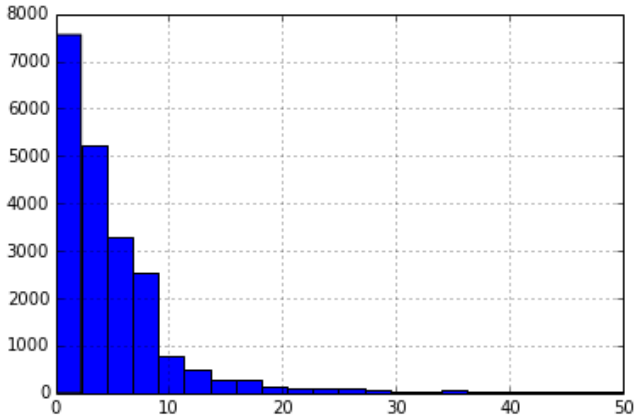
A kinyert adatbázis összesen 22.051 rekordot és 28 változót (mezőt) tartalmaz.

1. tábla: Kitöltött rekordok száma változónként

AJTSZAMA	21759	ELJ_TIP	21912
AKC_ISZ	21477	EREDMENYES	17433
AKC_T	21477	FELAD_D	21923
AKC_UT	21477	ID	22051
AKN	21798	KOZZT_D	17621
AKTIP	21798	KVETEL_D	21905
ALV	279	NYC	18641
ATC	20820	NYN	21922
ATN	21884	NY_SZ	18641
AT_SZ	20820	NY_TIP	18641
AT_TIP	20820	OSSZEG	18792
EH_D	17379	TARGY	21888
ELBSZ	4382	T_raw	21888
ELB_D	4399	URL	22051

A hiányzó adatok legtöbb esetben az eredményhirdetésről hiányoznak, nem azért mert nem sikerült a feldolgozás.

1. ábra: A beérkezett ajánlatok számának megoszlása:



2. tábla: Az ajánlatkérők, ajánlatok és nyertesek város szerint (top10)

Ajánlatkérő		Ajánlatok		Nyertes	
BUDAPEST	11615	BUDAPEST	51604	BUDAPEST	20746
MISKOLC	660	SZEGED	2260	DEBRECEN	789
GYŐR	599	DEBRECEN	2193	MISKOLC	764
DEBRECEN	581	MISKOLC	2084	GYŐR	691
SZEGED	575	SZÉKESFEHÉRVÁR	1934	SZEGED	634
PÉCS	521	GYŐR	1868	SZÉKESFEHÉRVÁR	621
KECSKEMÉT	311	NYÍREGYHÁZA	1704	NYÍREGYHÁZA	528
SZOMBATHELY	293	ZALAEGERSZEG	1623	PÉCS	485
NYÍREGYHÁZA	276	PÉCS	1561	GÖDÖLLŐ	482
SZÉKESFEHÉRVÁR	240	KECSKEMÉT	1169	TÖRÖKBÁLINT	379

3. Tábla: Ajánlatok és nyertesek név szerint (top10)

Ajánlatok		Nyertes	
STRABAG ÉPÍTŐ KFT.	569	JOHNSON & JOHNSON KFT.	381
JOHNSON & JOHNSON KFT.	401	B.BRAUN MEDICAL KFT.	314
SIEMENS RT.	374	HUMANTRADE KFT.	226
BETONÚT RT.	356	HUNGAROPHARMA RT.	225
MASZER RT.	348	ALLEGRO KFT.	164
B.BRAUN MEDICAL KFT.	347	STRABAG ÉPÍTŐ KFT.	164
HOFFMANN RT.	342	SIEMENS RT.	162
STRABAG RT.	315	INTERIMPORT KFT.	151
BETONÚT SZOLGÁLTATÓ ÉS ÉPÍTŐ RT.	315	EUROMEDIC PHARMA RT.	125
MÉLYÉPÍTŐ BUDAPEST KFT.	313	DIAGNOSTICUM RT.	125

Top 10 ajánlatkérő	
Magyar Távközlési Rt.	282
Magyar Turizmus Rt.	200
Budapesti Elektromos Művek Rt.	195
Országos Egészségbiztosítási Pénztár	194
Semmelweis Egyetem	178
Honvédelmi Minisztérium	156
Nemzeti Szakképzési Intézet	151
Kincstári Vagyoni Igazgatóság (KVI)	140
Adó- és Pénzügyi Ellenőrzési Hivatal	126
Budapest Főváros Önkormányzata Főpolgármesteri Hivatal	126

Az esetek 47%-ban a nyertes és ajánlatkérő városa megegyezik.

Az esetek 43%-ban az ajánlattevők és ajánlatkérők városa megegyezik.