

Diagnosztikai adatok kinyerése és hasznosítása okosautós környezetben

Dr. Ekler Péter, Dr. Lengyel László, Sik Dávid

ekler.peter@aut.bme.hu, lengyel.laszlo@aut.bme.hu, siktdavid@gmail.com

Automatizálási és Alkalmazott Informatikai Tanszék,

Budapesti Műszaki és Gazdaságtudományi Egyetem, 1111 Budapest, Műegyetem rkp. 3.

Absztrakt—A járművek manapság már nemcsak közlekedési vagy szállítási funkciókat szolgálnak. Ugyanolyan szenzorrendszernek, adatgyűjtő egységnek tekinthetők, mint például az okostelefonunk. Az adatok kinyerésére két fő interfészt használunk: az OBD-II portot (On-board Diagnostics II) és CAN bus-t (Controller Area Network). Megfelelő eszközökkel és okostelefonunkkal kinyerhetjük ezen interfészeken az adatokat, és bekapcsolhatjuk járművünket az Internet of Things (IoT) világába. A tanszéki fejlesztésű SensorHUB keretrendszerünkkel az összegyűjtött adatokból adatelemzési módszerekkel üzenet és adatfolyamokat készíthetünk, melyek alapján további következtetéseket vonhatunk le a vezető és a jármű tulajdonságairól. Az adatokat a telefonra visszaküldve segíthetjük a vezetői döntéshozatalt, valamint becsatlakozhatunk az okosautó-okosváros koncepció megvalósításába is. A cikk egy járműdiagnosztikai bevezetéssel kezd, majd bemutatásra kerül a SensorHUB keretrendszer és a ráépítő alkalmazások és komponensek.

Kulcsszavak—OBD, CAN, járműdiagnosztika, IoT, SensorHUB, VehicleICT, Big Data, Hadoop, üzleti intelligencia

I. BEVEZETÉS

Napjainkban az okos eszközök és a hozzájuk kapcsolódó szolgáltatások egyre jelentősebb szereppel bírnak a legtöbb iparág számára.

Az elmúlt évtizedekben nagyban nőtt az internetre csatlakozó eszközök száma, vagyis kiépült az Internet of Things (IoT). Ez főként az okostelefonok és a hordozható „kütyük” fejlődésének, növekedésének és elterjedésének köszönhető.

Ha egy okostelefonra nemcsak telefonként tekintünk, hanem szenzorok halmazaként, akkor könnyen beláthatjuk, hogy ezen adatok felhasználásával hihetetlen mennyiségű információ nyerhető ki.

Gondoljunk csak a földrajzi koordinátákra (mind GPS, mind a bázisállomások alapján), a mikrofonra, a kamerára, a gyorsulásmérőre, a fényérzékelőre, pulzusszámmérőre stb. És ezek csak a legalapvetőbb, szinte minden okostelefonban és okosórában elérhető mérőegységek.

2017-ben egy IP-alapú szenzor átlagos ára nagyságrendileg 500 Ft lesz, 2020-ban pedig 30 milliárd IP-szenzor lesz működésben. [1]

Ha kicsit absztraktabban gondolunk egy autóra, vagy bármilyen járműre, akkor könnyen beláthatjuk, hogy az is egy szenzorhalmaz. Annyiban tér el egy okostelefonról, hogy az adatai nem töltődnek fel az internetre, de ugyanúgy minden olvasható a műszerfalán, a kijelzőkön, mérhető az egyes

rendszerek feszültsége, árama, teljesítménye.

A szenzoradatokhoz kapcsolódási lehetőség például az OBD-II (On-board Diagnostics II) és a CAN bus (Controller Area Network bus) interfészek, amelyek segítségével a járművekből kinyerhetők különféle információk, pl. sebesség, fordulatszám, valamint különböző hőmérséklet és nyomásértékek is. Ezeket felhasználva pedig elfogadhatóan közelíthető számos hasznos információ, pl. a jármű aktuális fogyasztása, illetve CO₂-kibocsátása. [3]



1. ábra: Bluetooth-os OBD adapter

Az OBD-II port egy egyszerű adapterrel, Bluetooth-on vagy USB-n keresztül tud kommunikálni, adatfolyamot küldeni. Ezeket egy okostelefon segítségével már könnyen továbbítani lehet a szerveroldal irányába.

A kinyert és a származtatott adatokra alapozva, majd kiegészítve további funkciókkal különböző okosautó-alapú megoldások készíthetők a jármű használója, de akár közösségek vagy ipari résztvevők (flották, biztosítók, szervizek, stb.) számára.

II. ON-BOARD DIAGNOSTICS (OBD)

Az Amerikai Egyesült Államok Kaliforniában működő levegőtisztaság-védelmi hatósága (CARB) felismerte a folyamatos állapotfelügyelet jelentőségét, és a gyártók részére előírásban rögzítette a gépjárműemisszió-korlátozó műszaki rendszereinek fedélzeti ellenőrzési kötelezettségét. [4]

Az OBD-I néven ismertté vált fedélzeti diagnosztikai rendszert az 1988-as modellévtől kezdve kötelezővé tették. A szabályozás műszaki előírásait SAE (Society of Automobile Engineers) szabványok és ajánlások rögzítik.

Az OBD-I előírásokat az 1994-es modellévtől kezdődően felváltották az OBD-II előírások. Az OBD-II a

személygépjárművekre és a könnyű haszongépjárművekre, az 1996-os modellévtől kezdődően pedig a dízelmotorral meghajtott gépjárművekre is hatályos az USA-ban.

Az OBD-II európai megfelelője az EOBD, amelynek bevezetését az Európai Unió tagországaiban egy 2001-es irányelv írta elő.

Az OBD-II szabvány 9 különböző módot (szintet) definiál a diagnosztikai eszközök számára. Az OBD rendszert nem teljes körűen támogató járműveknél előfordulhat, hogy a jármű vezérlőegysége csak az első néhány módot képes megvalósítani.

A diagnosztika 9 szintje a következő [4] [5]:

- 1. mód: diagnosztikai adatok kiolvasása.
- 2. mód: Freeze Frame paraméterek (egy adott hibakód eltárolásakor mért paraméter információk).
- 3. mód: hibakód kiolvasás.
- 4. mód: hibakód tároló törlése.
- 5. mód: Lambda szonda tesztertek kijelzése.
- 6. mód: nem folyamatosan figyelt funkciók mérési értékei, vizsgálati és küszöbértékek kijelzése.
- 7. mód: hibakód kiolvasás.
- 8. mód: beavatkozó tesztek (Európában nem használatos).
- 9. mód: járműspecifikus adatok és információk.

Az 1. módot lehet használni a folyamatos, vezetés közbeni adatkiolvasásra, a további módok főleg a szervizelés folyamán kerülnek előtérbe.

Ha az autókban vezetés közben, vagy a gyújtás ráadása után a műszerfalon felvillan a MIL (Malfunction Indicator Light) indikátorlámpa, akkor abból valamilyen hibára következtethetünk. Ennek és a további hibajelzések feloldásáról elsősorban az autó üzemeltetői kézikönyvéből érdemes tájékozódni és megpróbálni elhárítani az okokat.

Összetettebb veszélyek és hibák esetén nem biztos, hogy találunk a kézikönyvben konkrét megoldást, ilyenkor célszerű egy szakszervizt felkeresni, és ott kiolvasatni a hibakódot.

A hibakód (DTC, Diagnostic Trouble Code) kiolvasása az eddig ismert OBDD rendszeren keresztül történik (3-as és 7-es mód), a kód jelentésének feloldása után lehetőség van a hibakód tárolásakor mért paraméterek kiolvasására (2-es mód), a hiba elhárítására, valamint a hibakód törlésére is (4-es mód).



2. ábra: OBD csatlakozó helye egy Rover 25-ös modellben

Az OBD-II szabvány szerint a hibakódok 5 karakterből állnak, amelyből az első karakter betű, a további négy karakter szám, pl. P0506.

Az előírások megkövetelik a gyártóktól az OBDD adatokhoz történő egységes hozzáférési lehetőséget. Ennek megfelelően a diagnosztikai csatlakozó kivitelét, annak elhelyezését, és az adatátvitelhez használható protokollokat is szabályok rögzítik.

Az aljzat beépítésére olyan helyet kell választani, ahol a karbantartó személy számára hozzáférhető, de a normál üzemeltetés során védve van. A gyakorlatban a csatlakozó a műszerfalban vagy alatta, a kormánykerék közelében, 1 méteren belül található. [5]

Az OBDD csatlakozást sokáig csak a szervizekben használták. 2005-ben az ELM Electronics készítette el az ELM327 mikrovezérlőt, aminek segítségével az OBDD-s jeleket a számítógépen elérhető soros RS-232 porton olvashatóvá alakítja. [6] Azóta már megjelentek az USB-s, valamint a Bluetooth-os és Wi-Fi-s átalakítók is.

Az OBDD port is a jármű CAN bus rendszerére csatlakozik, de rajta keresztül kevesebb szenzor érhető el, mert az újabbak nincsenek benne az eredeti szabványban. [3]

A CAN bus célja, hogy a járművekben található vezérlők és eszközök közvetlenül tudjanak kommunikálni egymással különböző üzenetekkel, egy soros buszrendszeren keresztül. Vagyis nem kell mindenkit-mindenkivel összekötni a kommunikációhoz, jelentősen csökkentve a felhasznált vezetékek mennyiségét, tömegét és árát, valamint a kommunikációs hibák felderítésének nehézségét. [7]

A CAN bus fejlesztése 1983-ban kezdődött meg a Bosch-nál. 1986-ban rögzítette a szabványt a SAE, 1987-ben pedig elkészült az első vezérlő az Intel és a Philips gyárában. 1991-ben jelent meg a CAN 2.0-ás változata, majd ezután is még számos protokoll-változat készült el, és ma már az ipar szinte összes területén alkalmazzák.



3. ábra: FMS Gateway szimulátor

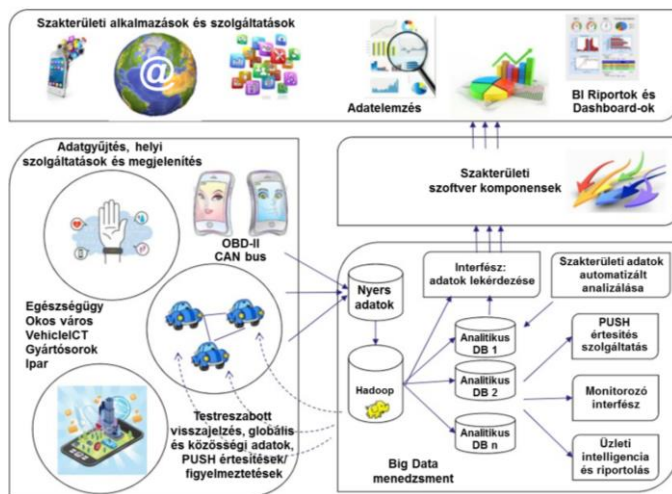
A CAN bus elérése és az adatok kinyerése már sokkal összetettebb és specifikusabb feladat. Ebben nyújt segítséget az Inventure Kft. által gyártott FMS Gateway eszköz, amely az adatokat CAN, RS-232 és Bluetooth-os kimenetre tudja továbbítani, buszokból, kisáruszállítókból, teherautókból és akár személyautókból is. [6]

III. SENSORHUB KERETRENDSZER

A Budapesti Műszaki és Gazdaságtudományi Egyetem Automatizálási és Alkalmazott Informatikai Tanszékén fejlesztett SensorHUB keretrendszer különböző szakterületek adatait tudja rögzíteni és feldolgozni. Alkalmazási területei lehetnek például az egészségügy, okos városok, járműipar, gyártósorok stb. Ezek tipikusan olyan területek, ahol nagy mennyiségű szenzoradatok feldolgozása, tárolása, elemzése és felhasználása a feladat.

A SensorHUB komponensei [2]:

- Szenzorok, adatok gyűjtése, helyi szolgáltatások, megjelenítés és adatátvitel,
- Felhő alapú háttérrendszer, Big Data menedzsment funkciókkal,
- Szakterület specifikus szoftver komponensek,
- Alkalmazások, szolgáltatások, üzleti intelligencia jelentések és elemzések.



4. ábra: A SensorHUB keretrendszer felépítése

A SensorHUB koncepciónak előnye, hogy a gyakran előforduló, adatokkal kapcsolatos műveleteket csak egyszer kell implementálni és ezek után megfelelő dokumentációkkal támogatva bárki tud olyan alkalmazást készíteni, ami felhasználva a SensorHUB keretrendszert, képes lesz adatai megjelenítésre, feltöltésre, elemzésére. [9]

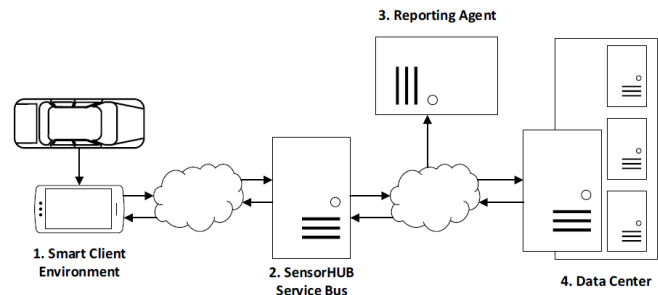
2015-ben a SensorHUB koncepció és keretrendszer elnyerte a Pro Progressio Alapítvány Innovációs Díját. [10]

A SensorHUB keretrendszer részeként készült VehicleICT alrendszer ezt a kapcsolatot teremti meg, vagyis az autónk mostantól okosautó lesz (még néhány további eszköz segítségével), és így az abból nyert adatok összemérhetőek lesznek és bekerülnek a többi okoseszköz által gyűjtött adathalmaz mellé. [2]

A VehicleICT rendszer a SensorHUB koncepciónak és keretrendszernek a járműipari szakterületre történő alkalmazása. Ezt a rendszert számos alkalmazás felhasználja, mint például a Social Driving és a nemrég elkészült ObdCanCompare.

Ez az első szakterületi alkalmazás, ami elkészült a

SensorHUB rendszerhez. A rendszer foglalkozik a járművekben lévő adatok kinyerésével, sorosításával, származtatott értékek számításával, kliens oldali megjelenítéssel, szerver oldali feltöltéssel, valamint ezen adatok elemzésével és további felhasználásával. [11]

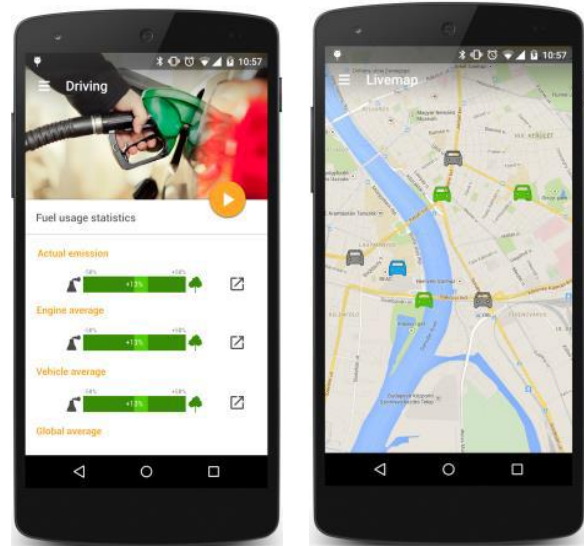


5. ábra: A VehicleICT architektúráis felépítése

A felhasználó eszközén futó szolgáltatás végzi az előbb leírt feladatokat. Magának a szolgáltatásnak nincsen felhasználói felülete, hanem egy alkalmazásba beépítve, androidos osztálykönyvtárként működik.

IV. SOCIAL DRIVING

A Social Driving alkalmazással a járműből kinyert adatok segítségével össze lehet kapcsolni a rendszer felhasználóit, így összehasonlítható lesz vezetési stílusuk, és ha ugyanolyan autót vezetnek, akkor akár meg lehet nézni, milyen eltérések mutatkoznak a fogyasztásban a különböző vezetési stílusok eredményeként. [12]



6. ábra: A Social Driving alkalmazás felhasználói felülete

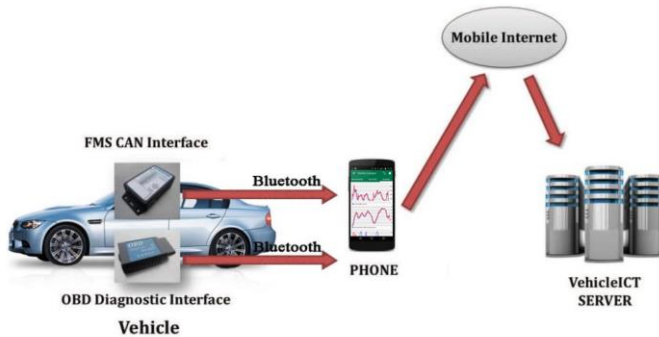
Az adatok felhasználása számos üzleti lehetőséget is rejt. Például jutalmazni lehet azokat, akiknek autója adott idő alatt a legkevesebb káros anyagot bocsájtja ki; ők a töltőállomásokon akár kedvezményt is kaphatnának tankolásakor. Még arra is lesz lehetőségük, hogy

utánanézzenek, az övékével megegyező autó mennyit fogyaszt, vagy milyen menettulajdonságokkal rendelkeznek.

A hibakódok kiolvasásával és valós idejű értesítéssel, a szervizhálózatok remekül használhatnák ezeket az információkat. Jelentős hiba esetén azonnal értesíteni lehet majd az autó vezetőjét, sőt akár online motordiagnosztikát is elvégezhetnek, hiszen rendelkeznek a szükséges adatokkal, így már felkészülten fogadhatnák az autót a szervizben, és akár személyre szabott ajánlattal is megkereshetik a tulajdonosokat. Ha pedig összekapcsoljuk ezt a rendszert az okos városéval, akkor valós időben jelezhető az autósoknak, ha már fölösleges a gázba taposnia, hiszen a kereszteződési lámpa mindjárt pirosra fog váltani. [13]

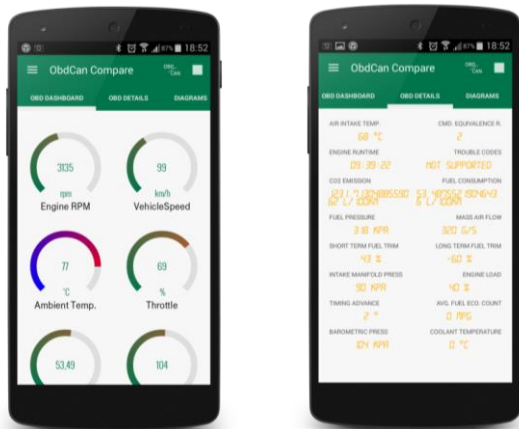
V. OBDSCANCOMPARE

Az ObdCanCompare az előző fejezetekben bemutatott OBD-II és CAN bus mérést egyszerre megvalósító és összehasonlítását biztosító, a SensorHUB keretrendszerrel felhasználó Android platformra készült alkalmazás.



7. ábra: A mérés lebonyolításának megvalósítási módja

Az alkalmazásban először a mérések tulajdonságait állíthatjuk be (a Bluetooth engedélyezése után): a két Bluetooth-os eszközt, az üzemanyag típusát, a motor térfogatát és hatásfokát.



8. ábra: Az alkalmazás képernyői (Dashboard, Details)



9. ábra: Az alkalmazás képernyői (Charts)

Ezután indíthatjuk el a mérést. Ekkor mind a Dashboard, mind a Details, mind a Diagrams füleken olvashatóak az adatok. Az OBD és CAN mérés adatainak megjelenítés között az OBD/CAN gombbal lehet váltani.

Mérés közben, ha van internetkapcsolat, akkor a mért adatok feltöltésre kerülnek a SensorHUB szerverére. Továbbá, ha a helymeghatározás is engedélyezve van, akkor az adatsomagokba bekerülnek a mérés közben aktuális földrajzi koordináták is.



10. ábra: Tesztelés

VI. BIG DATA A SZERVER OLDALON

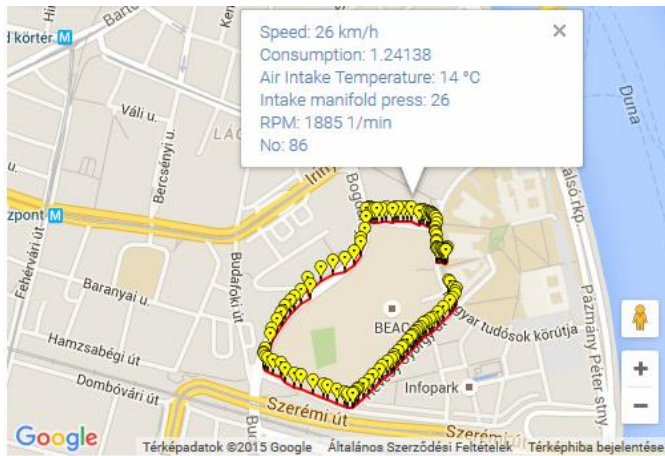
A szerveroldal alapja az elosztott fájlrendszer, a Cloudera Hadoop rendszere, melynek Data Lake-jében gyűlhetnek a telefon szenzorjaiból az adatok. Az adathalmaz feldolgozásához és elemzéséhez a Pentaho Data Integration és Pentaho Report Designer alkalmazások használhatóak például. A Pentaho egy nyílt forráskódú, Java alapú, Big Data rendszerekkel összeköthető programcsomag. [14]

A Data Integration különböző forrásokból származó adatokat tud egységesen kezelni. ETL (Extract-Transform-Load) transzformációkat állíthatunk össze benne a Spoon eszköz grafikus felületén. Ez a rész az adatfolyam manipulálásáról szól, ekkor kell kiválasztani azokat a rekordokat és attribútumokat, melyekre az elemzés során szükségünk lesz. [15]



11. ábra: Pentaho Data Integration adatfolyam

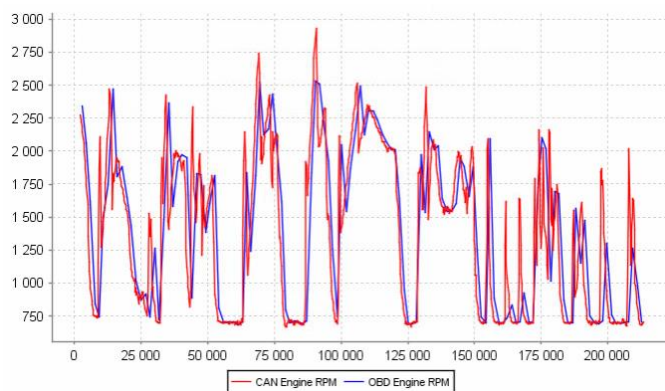
Az így elkészült adattáblát a Report Designer-ben lehet felhasználni különböző táblázatok, grafikonok, diagramok és további megjelenítők adatforrásként. Ennek kimenetei különböző formátumúak lehetnek: PDF, HTML, Excel, RTF, Text, CSV. Továbbá a struktúrák feltölthetők az üzleti intelligencia szerverre, ami generálni tudja a megfelelő formátumú riportot.



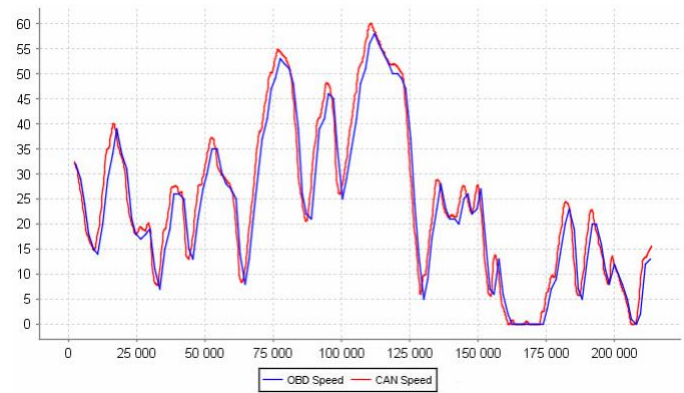
12. ábra: Interaktív és statikus térkép megjelenítés a Pentaho riportban

A térképes megjelenítés nincs beépítve a Report Designer-be, ezt egy a riportba belekódolt Javascript kódrészlet hozza létre, mely a Google Maps API adatformátumát rendeli össze a lekérdezés eredményének soraival.

Az előző mobilképernyős és a következő ábrából kitűnik, hogy mennyire hasonlítanak az ObdCanCompare-ben látható diagramok, a Pentaho segítségével készült diagramokhoz.



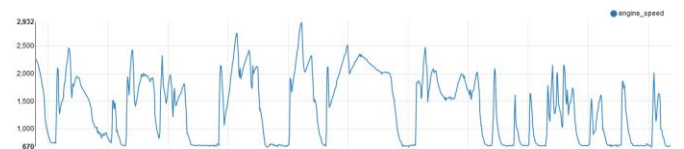
13. ábra: Fordulatszám összehasonlítása a Pentaho riportban



14. ábra: Sebesség összehasonlítása a Pentaho riportban

Tehát a Pentaho segítségével is szépen ki lehet rajzoltatni ugyanazokat a diagramokat.

A két megjelenítési forma tehát konzisztens. Az ObdCanCompare-es az azonnali megjelenítéshez hasznos, míg a Pentaho-s pedig a későbbi elemzésekhez szolgál célt.



15. ábra: Fordulatszám a HUE felület diagramján

A Hadoop klaszter online HUE felülete is számos adatvizualizációs lehetőséget biztosít, de csak az egyszerűbbek közül. [16] Az adatfolyam összeállítására itt az Oozie eszközt lehet használni, amiben folyamatábraként rakhatjuk össze az ETL folyamatot. [17] Ezt lehet a szerveren időzíteni is, például, hogy milyen időközönként fusson le.



16. ábra: HUE Map a sebesség kijelzésével

A térképes megjelenítést az online felületen sokkal egyszerűbb összerakni, nem kell kódot írni, csak ki kell választani a földrajzi hosszúságnak és szélességnek megfelelő oszlopokat. Ehhez hozzárendelhető, hogy a térképen rákattintva mit írjon ki az adott ponthoz, de a koordinátákhoz csak egy paramétert lehet kiírni. Végül az eredmény megjelenik egy OpenStreetMap alapú térképen.

VII. ÖSSZEGZÉS, JÖVŐBELI LEHETŐSÉGEK

Az elkészült riportok és diagramok alapján látható, hogy az OBD-II mérés és a CAN bus mérés eredményei számos jellemzőben szinte megegyeznek egymással. Ilyen például a fordulatszám és a sebesség értéke. Itt csak egy apró késleltetés látható az OBD-s adatokban a CAN bus adataihoz képest, amit könnyen lehet korrigálni. Ugyanakkor más, főleg a származtatott (aggregált) értékek már nagyobb eltéréseket mutattak. Itt az eltérés oka, hogy különböző összetett formulák léteznek például a fogyasztás vagy a CO₂-kibocsájtás meghatározására, amikből eltérő eredmények születhetnek az OBD-s és CAN bus-os mérés összehasonlításában.

Ezek alapján elmondható, hogy az alapvető okosautó-alapú megoldásokhoz elegendő az OBD-II port használata, például a Social Driving alkalmazás esetében, de fontos, hogy a származtatott értékek formulái mindenütt egységesen kerüljenek felhasználásra. Amennyiben pedig professzionális megoldásra van szükségünk, például flottakövetésre, és további összetettebb adatok kinyerésére, akkor a CAN bus-alapú megoldás a jó választás.

Természetesen ezek a mérések és adatgyűjtések csak a legegyszerűbb, a működést szemléltető példák voltak. Célunk a jövőben, hogy a beérkező adathalmaz alapján összetettebb üzleti intelligencia módszerekkel további érdekes és hasznos összefüggéseket nyerjünk ki, valamint a tendenciák alapján a jövőbeli értékeket jelezzük előre. Ezekkel az adatokkal akár a gépjármű vezetőjét is tudjuk támogatni, ha visszaküldjük őket az okostelefonra és így bekapcsolódhatunk a mostanában egyre inkább feltörekvő okosautó-okosváros koncepció megvalósításába is.

KÖSZÖNETNYILVÁNÍTÁS

A kutatás folytatása és a publikáció az Emberi Erőforrások Minisztériuma által az Új Nemzeti Kiválóság Program keretében meghirdetett Felsőoktatási Mesterképzés Hallgatói Kutatói Ösztöndíj, valamint a Magyar Tudományos Akadémia Bolyai János Kutatói Ösztöndíj támogatásával jött létre.

IRODALOMJEGYZÉK

- [1] Charaf Hassan: *Az infoszféra tudást közvetítő szerepe a mai társadalomban* <http://www.matud.iif.hu/2015/02/03.htm> (2015. október)
- [2] SensorHUB keretrendszer <https://www.aut.bme.hu/Pages/Research/SensorHUB> (2015. október)
- [3] VehicleICT keretrendszer <https://www.aut.bme.hu/Pages/Research/vehicleict> (2015. október)
- [4] Lakatos István, Nagyszokolyai Iván: *Gépjárművek üzeme I.* http://www.tankonyvtar.hu/hu/tartalom/tamop412A/0018_Gepjarmuvek_uzeme_1/adatok.html Typotex Kiadó, 2012 (2015. október)
- [5] Menedzsment és motordiagnosztika <http://www.geree.hu/motormenedzsment-es-diagnosztika/> (2015. október)
- [6] ELM327 <http://www.elmelectronics.com/obdic.html#ELM327> (2015. október)

- [7] Dr. Fodor Dénes, Dr. Szalay Zsolt: *Autóipari kommunikációs rendszerek* http://moodle.autolab.uni-pannon.hu/Mecha_tananyag/autoipari_kommunikacios_rendszerek/ch03.html (2015. november)
- [8] CAN bus communication <http://fmsgateway.com/glossary/can-bus-communication> (2015. november)
- [9] László Lengyel, Péter Ekler, Tamás Ujj, Tamás Balogh and Hassan Charaf: *SensorHUB: An IoT Driver Framework for Supporting Sensor Networks and Data Analysis* International Journal of Distributed Sensor Networks, 2015. <http://www.hindawi.com/journals/ijdsn/2015/454379/> (2015. október)
- [10] Innovációs díjban résztült a SensorHUB <https://www.vik.bme.hu/hir/874-bme-innovacios-dijban-reszesult-a-sensorhub-fejlesztes> (2015. október)
- [11] Jereb László, Lengyel László: *Jármű ICT fejlesztési irányok és kihívások* Híradástechnika, HTE Infokom 2014 – LXIX. évfolyam, 2014.
- [12] László Lengyel, Péter Ekler, Tamás Ujj, Tamás Balogh and Hassan Charaf: *Social Driving in Connected Car Environment*, European Wireless 2015.
- [13] A SensorHUB, Mérnök újság, XXII. évf. 10. szám, 2015. október
- [14] Pentaho Community <http://community.pentaho.com/> (2015. november)
- [15] Dudás Ákos, Ekler Péter, Tömösvári Imre: *Üzleti intelligencia* tantárgyi jegyzetek <https://www.aut.bme.hu/Course/BMEVIAUMA02> (2015. november)
- [16] HUE – The Hadoop UI <http://gethue.com/> (2015. december)
- [17] Oozie Workflow Scheduler <http://oozie.apache.org/> (2015. december)