DOI:10.15774/PPKE.ITK.2019.003

eπQue: Gépi fordítás minőségét becslő programcsomag

Doktori (PhD) disszertáció

Yang Zijian Győző

Roska Tamás Műszaki és Természettudományi Doktori Iskola
Pázmány Péter Katolikus Egyetem
Információs Technológiai és Bionikai Kar

Témavezető
Dr. Prószéky Gábor

Budapest, 2019

DOI:10.15774/PPKE.ITK.2019.003

„[A bölcsesség] birtokába jutni jobb, mint ezüstöt szerezni, aranynál többet ér ennek elérése.”

– Péld 3,14

DOI:10.15774/PPKE.ITK.2019.003

Köszönetnyilvánítás

Mindenekelőtt szeretnék köszönetet mondani témavezetőmnek, Dr. Prószéky Gábornak, akitől mind szakmailag, mind emberileg számtalan támogatást kaptam az elmúlt évek során. Köszönöm Neki, hogy mindvégig baráti közvetlenséggel fordult felém. Nélküle ez a munka nem jöhetett volna létre.

Külön köszönetet szeretnék mondani Dr. Laki Lászlónak, aki fáradhatatlanul végigkísért a doktori éveim alatt. Szakmai és baráti támogatással dolgoztunk együtt és adott útmutatást. Nélküle nem jöhettek volna létre a kutatásaim.

Köszönetet szeretnék mondani kutatótársaimnak, Dr. Siklósi Borbálának és Dömötör Andreának, akikkel közösen hoztunk létre értékes munkákat. Nem mellesleg számtalanszor voltak nyelvi lektoraim.

Köszönöm a közeli munkatársaimnak, a 314-es szoba volt és jelenlegi munkatársainak, Dr. Novák Attilának, Kalivoda Ágnesnek, Ligeti-Nagy Noéminek, Vadász Noéminek, Dr. Indig Balázsnak, Dr. Orosz Györgynek, Dr. Miháltz Mártonnak és Dr. Endrédy Istvánnak az inspiráló beszélgetéseket, a közös gondolkodásokat, a sok segítséget és a vidám légkört.

Köszönettel tartozom a Tanulmányi Osztály és a Gazdasági Osztály munkatársainak, valamint a könyvtárosoknak az évek során nyújtott segítségért. Külön köszönetet mondok Dr. Vida Tivadarnénak, aki mindig szívélyes barátsággal bíztatott és segített az előremenetelemben.

Köszönöm Szulyovszky Dávidnak és Bolgári Csabának a nyelvi lektorálást.
Köszönetemet fejezem ki a MorphoLogic Lokalizáció Kft. támogatásáért, hogy lehetővé tette számomra a korpuszok használatát a kutatásomhoz. Munkám részben a magyar kormány EFOP-3.6.3-VEKOP-16-2017-00002 pályázati programjának támogatásával, az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásában valósult meg.

Végül, a legfontosabb, szeretném megköszönni szeretett feleségemnek, Patríciának és az egész családomnak az évek során nyújtott biztatást, segítséget, türelmet, és hogy minden lehetséges módon támogattak kutatásaim alatt.

DOI:10.15774/PPKE.ITK.2019.003

Abstract

Machine translation has become a daily used tool among people and companies. There are significant differences in the quality of machine translation systems. Thus the measurement of the quality of translation output has become an important research field in natural language processing.

A quality score for machine translation could save a lot of time and money for users, companies and researchers. Knowing the quality of machine translated segments can help human annotators in their post-edit tasks, or using the quality we can detect the errors in a translation, or filter out and inform users about unreliable segments. Additionally, quality indicators can help machine translation systems to combine the translations to produce higher quality at the system level.

In my research, I used the quality estimation method for three different tasks.

First, I created a quality estimation system for English-Hungarian machine translation, that has not been implemented before by others. To build the English-Hungarian quality estimation system, I created a training corpus. Then, I did experiments in feature engineering. As a first step, I implemented the given features for Hungarian, then I created new semantic features. Using these semantic features, I could gain better results than the baseline feature set. I also created optimized feature sets, which produced further improvement in results.

In my second task, using the English-Hungarian quality estimation system, I created a composite machine translation system, which combines the output of multiple machine translation systems. Results showed that my composite system gained better final translation quality compared to the translation systems alone. My combination method was tested on various language pairs.

Finally, I used the quality estimation method for monolingual quality estimation. I built an error analyser system, that can detect human made mistakes or errors.

DOI:10.15774/PPKE.ITK.2019.003

Kivonat

A gépi fordítás használata mára széles körben elterjedt a hétköznapokban és a cégek életében, azonban a létező rendszerek fordítási minőségében jelentős különbségek mutatkoznak. Ezért a gépi fordítás minőségének becslése fontos kutatási terület a nyelvtechnológiában. Cégek esetében egy minőségi mutató segítséget nyújthat a gépi fordítás utómunkáját végző szakemberek számára. Másik alkalmazási területe a gépi fordítórendszerek kombinációja. Megfelelő minőségbecsléssel több gépi fordítórendszer kimenetét kombinálva javíthatjuk a rendszerünk minőségét. Végül, ismerve a fordítás minőségét, ki tudjuk szűrni a hibás vagy használhatatlan fordításokat.

A kutatásom során a minőségbecslés módszerét alkalmaztam három különböző feladatra. Először létrehoztam egy minőségbecslő rendszert, amit angol-magyar nyelvre azelőtt más nem készített. Létrehoztam egy tanító korpuszt, amelynek segítségével betanítottam a minőségbecslő rendszeremet. A tanításhoz szükség volt minőségbecslő jegyekre. A munkámban kipróbáltam azokat a jegyeket, amelyeket mások készítettek. Ezt követően angol-magyar nyelvre létrehoztam saját szemantikai jegyeket, amelyekkel eredményjavulást értem el az alapjegykészlethez képest. Optimalizációt is végeztem, amellyel úgy értem el további eredményjavulást, hogy csak a releváns jegyeket használtam.

Második feladatként a létrehozott angol-magyar minőségbecslő rendszer segítségével különböző gépi fordítórendszerek kimenetét kombináltam. Ezzel egy kompozit fordítórendszert hoztam létre, amelynek rendszerszintű minősége jobb, mint az általa felhasznált gépi fordítórendszerek önmagukban. A módszert teszteltem több nyelvpárra is, és mindegyik általam kipróbált nyelvpárra érvényesült az előzőleg megfogalmazott eredmény.

Végül, a minőségbecslő módszert egynyelvű szövegek minőségének meghatározására próbáltam ki. Létrehoztam egy hibaelemző rendszert, amelynek segítségével egynyelvű szövegekben fellelhető ember által vétett hibákat tud detektálni.

DOI:10.15774/PPKE.ITK.2019.003

Tartalomjegyzék

DOI:10.15774/PPKE.ITK.2019.003

DOI:10.15774/PPKE.ITK.2019.003

DOI:10.15774/PPKE.ITK.2019.003

DOI:10.15774/PPKE.ITK.2019.003

Ábrák jegyzéke

DOI:10.15774/PPKE.ITK.2019.003

DOI:10.15774/PPKE.ITK.2019.003

Táblázatok jegyzéke

DOI:10.15774/PPKE.ITK.2019.003

DOI:10.15774/PPKE.ITK.2019.003

DOI:10.15774/PPKE.ITK.2019.003

Rövidítésjegyzék

BLEU BiLingual Evaluation Understudy

CCI Corrected Classified Instances - Helyesen osztályozott egyedek

CoMT Composite Machine Translation system - Kompozít Gépi fordítórendszer

HBSMT Hierarchical-based Statistical Machine Translation - Hierarchikus Statisztikai gépi fordítás

HTER Human-targeted Translation Error Rate

LSA Latent Semantic Analysis - Látens szemantikai analízis

MAE Mean Absulute Error - Átlagos abszolút hiba

MT Machine Translation - Gépi fordítás

NMT Neural Machine Translation - Neurális gépi fordítás

oBLEU Ortho BLEU - Karakteralapú BLEU

oTER Ortho TER - Karakteralapú TER

PBSMT Phrase-based Statistical Machine Translation - Kifejezésalapú Statisztikai gépi fordítás

QE Quality Estimation - Minőségbecslés

RBMT Rule Based Machine Translation - Szabályalapú gépi fordítás

DOI:10.15774/PPKE.ITK.2019.003

RMSE Root Mean Squared Error - Szórás

RNN Recurrent Neural Network - Rekurrens Neurális Hálózat

SMT Statistical Machine Translation - Statisztikai gépi fordítás

SVM Support Vector Machine - Szupport Vektor Gépek

SVR Support Vector Regression - Szupport Vektor Regresszió

TER Translation Error Rate

DOI:10.15774/PPKE.ITK.2019.003

1. fejezet

Bevezetés

A gépi fordítás alkalmazása széles körben elterjedt mind a vállalatok, mind a magánszemélyek körében. Egyre nagyobb az igény a jó minőségű gépi fordítórendszerek iránt, azonban a különböző gépi fordítórendszerek között minőségben nagy eltérések lehetnek.

A gépi fordítás nehézsége abban rejlik, hogy míg egy ember több évnyi tudással, tapasztalattal, kreativitással, asszociációs képességekkel és egyéb tulajdonságokkal rendelkezik, addig egy gép csak azzal, amire megtanítjuk. A betanítás komoly kihívás, számtalan problémával jár. Ilyen például a szavak jelentésbeli, illetve szerkezeti többértelműsége, a különböző nyelvtani szerkezetekkel járó problémák, a szórendek stb. Az elmúlt évtizedek során nagy utat jártak be a gépi fordítórendszerek, és jelentős változásokon mentek keresztül. Manapság egyre több magánszemély és vállalat használja a gépi fordítószoftvereket. Mind a magánszemélyek, mind a cégek számára nagy segítséget nyújthat egy jó minőségű gépi fordítórendszer, azonban számtalanszor tapasztaljuk, hogy a gépi fordító gyenge minőségű fordítást állít elő. Egy adott fordítás minőségének automatizált módszerekkel történő meghatározása komoly kihívást jelent, egyúttal a gépi fordítás elterjedésével egyre több helyen merül fel igényként a gép által lefordított szövegek minőségének meghatározása. Cégek esetében igen nagy segítséget nyújthat egy minőségi mutató, amely a gépi fordítás utószerkesztését (post-edit) végző szakemberek munkáját támogatja és gyorsíthatja, illetve a fordítócégeket segítheti költségeik csökkentésében. Alkalmazható továbbá egy olyan minőségi mérőszám létrehozására, amellyel több gépi fordítómódszer fordítását lehet összehasonlítani, és a jobb fordítást kiválasztva, javít-

DOI:10.15774/PPKE.ITK.2019.003

hatják rendszerük végső minőségét és hatékonyságát. Végül, de nem utolsósorban, ha ismerjük a fordítás minőségét, akkor kiszűrhetjük a használhatatlan fordításokat, illetve figyelmeztethetjük a végfelhasználót a megbízhatatlan szövegrészekre.

A gépi fordítás kiértékelése a nyelvtechnológiai kutatásokban komolyabb figyelmet kapott az elmúlt években. Amikor a gép által fordított szöveg kiértékeléséről beszélünk, megkülönböztetjük az ember és a gép által történő kiértékelést. Az emberi kiértékelés a legpontosabb , ezért minden gépi módszer emberi kiértékelésen alapszik.

Alapvetően kétféle gépifordítás-kiértékelési módszert különböztethetünk meg. Az első a referenciafordítással történő kiértékelés, amelyet gyakran hagyományos módszernek is szokás hívni. A másik módszer a referenciafordítás nélkül történő kiértékelés, más néven minőségbecslés.

Disszertációmban a minőségbecslés módszerét alkalmaztam, három különböző feladatra. Először létrehoztam egy angol-magyar minőségbecslő rendszert, amit azelőtt más nem készített. Ehhez létrehoztam egy tanító korpuszt, majd ennek segítségével betanítottam a minőségbecslő rendszeremet. A tanításhoz szükség volt minőségbecslő jegyekre. A jegyek előállításához különböző kísérleteket végeztem. Kipróbáltam azokat a jegyeket, amelyeket más nyelvpárokra optimalizáltak, majd angol-magyar nyelvre létrehoztam saját jegyeket, és ezekkel optimalizációkat is végeztem.

Második feladatként a létrehozott angol-magyar minőségbecslő rendszer segítségével különböző gépi fordítórendszerek kimenetét kombináltam, és ezzel egy kompozit fordítórendszert hoztam létre, amelynek minősége jobb, mint a kombinált rendszerek önmagukban. A módszert kipróbáltam több nyelvpárra, és mindegyik általam kipróbált nyelvpárra érvényesült az előzőleg megfogalmazott eredmény.

Végül a minőségbecslő módszert egynyelvű szövegek minőségének meghatározására próbáltam ki. Létrehoztam egy hibaelemző módszert, amelynek segítségével egynyelvű szövegekben fellelhető hibákat tudtam detektálni.

A fenti három részből egy minőségbecslő programcsomagot állítottam össze, amelynek az eπQue nevet adtam

DOI:10.15774/PPKE.ITK.2019.003

2. fejezet

Elméleti háttér

2.1. Gépi fordítás

A gépi fordítás (Machine Translation - MT) tudománya [15] egyidős a számítógépek megjelenésével, és mind a mai napig igen fontos kutatási terület a számítógépes nyelvészetben. Az elmúlt évtizedekben a gépi fordítás jelentős mértékben fejlődött, számos szoftveres módszer született a természetes nyelvek közötti fordítás megoldására [16]. Manapság a gépi fordítás széles körben elterjedt a hétköznapokban is, egyre többen használják napi szinten[1]. Mind a magánszemélyek, mind a cégek körében egyre fontosabb a jó minőségű gépi fordítás alkalmazása. Azonban a különböző gépi fordítórendszerek között módszertanilag és minőségben is nagy eltérések lehetnek, ezért fontos ismerni az egyes gépi fordítási módszer működését. A következő alfejezetekben röviden bemutatom a különböző gépi fordítási módszereket.

2.1.1. Szabályalapú gépi fordítás

A szabályalapú gépi fordító (Rule Based Machine Translation) [16] beépített szótár és nyelvtani – főként szintaktikai és morfológiai – szabályok alapján végez közvetlen fordítást. Alapvetően három különböző módszert különböztetünk meg:

- *Direkt fordítás* (Direct Machine Translation): Legegyszerűbb formája a szabályalapú gépi fordításnak. A rendszer egy szótár segítségével szóról szóra fordítja le a forrásnyelvi szöveget a célnyelvre, majd a végén, a célnyelvi oldalon, szórendi

[1] https://blog.google/products/translate/ten-years-of-google-translate

DOI:10.15774/PPKE.ITK.2019.003

átalakításokkal javítja az eredményt. A módszer előnye, hogy könnyen megvalósítható, hátránya, hogy nem képes komplex nyelvtani szerkezeteket kezelni, ezért sok esetben rossz minőségű fordítást állít elő.

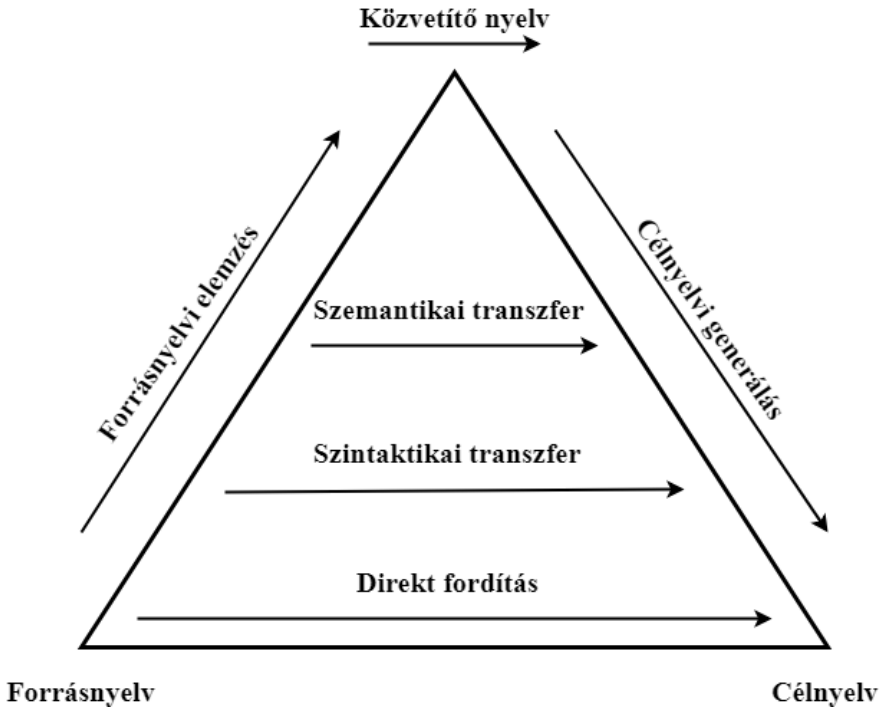
- *Transzfer fordítás* (Transfer-Based Machine Translation): A direkt fordítás módszerénél jobb fordítást eredményez, ha a szótár és az átrendezési szabályok mellé nyelvtani szabályokat is használunk. A forrásnyelvi szövegen különböző komplexitású szabályok segítségével elemzést végzünk (elemzés), majd az elemzett szöveget lefordítjuk a célnyelvre (transzfer), végül újabb előre definiált szabályok segítségével állítjuk elő (generálás) a helyes célnyelvi mondatot.

- *Közvetítőnyelves fordítas* (Interlingua Machine Translation): A közvetítőnyelves fordítás esetében a forrásnyelvet először egy köztes absztrakt reprezentációba képezzük le, ami az interlingua, majd ebből a közvetítő reprezentációból állítjuk elő a célnyelvi fordítást.

A szabályalapú gépi fordítás működését jól jellemzi a Vauquois-háromszög (lásd 2.1. ábra). Minél több az elemzés és a generálás szintje, annál jobb minőségű fordítást tudunk előállítani. A mélység alatt azt értem, hogy minél többféle elemzést végzünk magán a szövegen. Az elemzés lehet morfológiai, szintaktikai vagy szemantikai elemzés.

A szabályalapú gépi fordítórendszerekhez (főleg a transzfer és a közvetítőnyelves módszereknél) szükségünk van különböző nyelvi elemző rendszerekre. Minél jobb minőségű elemzők állnak a rendelkezésünkre, annál pontosabb fordítást tudunk készíteni. A szabályalapú módszereknek az egyik legnagyobb hátránya, hogy a fordítás minősége erősen függ az elemzők minőségétől, ugyanakkor az egyes típusú elemzések önmagukban is kutatási területek. Ha nem áll rendelkezésünkre jó minőségű elemző, a gépi fordításunk minősége is gyenge lesz. A másik probléma, hogy ezek az elemzők nyelvspecifikusak, ezért megnehezítik a rendszer számára az újabb nyelvre való kiterjesztést. Végül, ha a szótár kicsi, a fordítás fedése általában alacsony, vagyis kevés dolgot tud lefordítani. A szabályalapú gépi fordítás módszer előnye, hogy meglévő elemző rendszerek mellett magas pontosságot eredményez.

DOI:10.15774/PPKE.ITK.2019.003



2.1. ábra Vauquois-háromszög

2.1.2. Példaalapú gépi fordítás

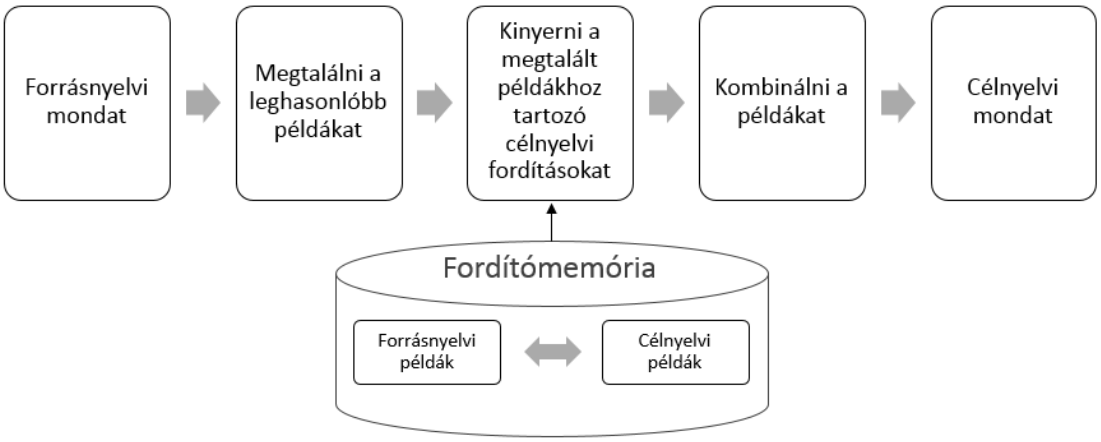
A példaalapú gépi fordítás (Example Based Machine Translation) [17] alapja az, hogy a már emberek által lefordított szövegeket felhasználjuk.

A példaalapú gépi fordítás középpontjában a fordítómemória áll. Az előre lefordított szövegszegmenseket, mint példákat tároljuk el a fordítómemóriában. A rendszer ezekből választja ki a fordítandó szöveghez a hozzá leghasonlóbb példákat (kifejezéseket vagy mondatokat), majd a kiválasztott részek egyesítésével és kombinálásával állítja elő a fordítást (lásd 2.2. ábra).

A fordítómemória önmagában alkalmazható más módszereknél is, mint például szabályalapú, vagy statisztikai gépi fordítórendszerek minőségének javítására.

A rendszer előnye, hogy azokat a szövegrészeket, amelyek szerepelnek a fordítómemóriában, magas pontossággal tudja lefordítani. Azonban, ha egy szövegrészlet nincsen benne a fordítómemóriában, a rendszer pontatlanul, vagy egyáltalán nem kezeli azt.

DOI:10.15774/PPKE.ITK.2019.003



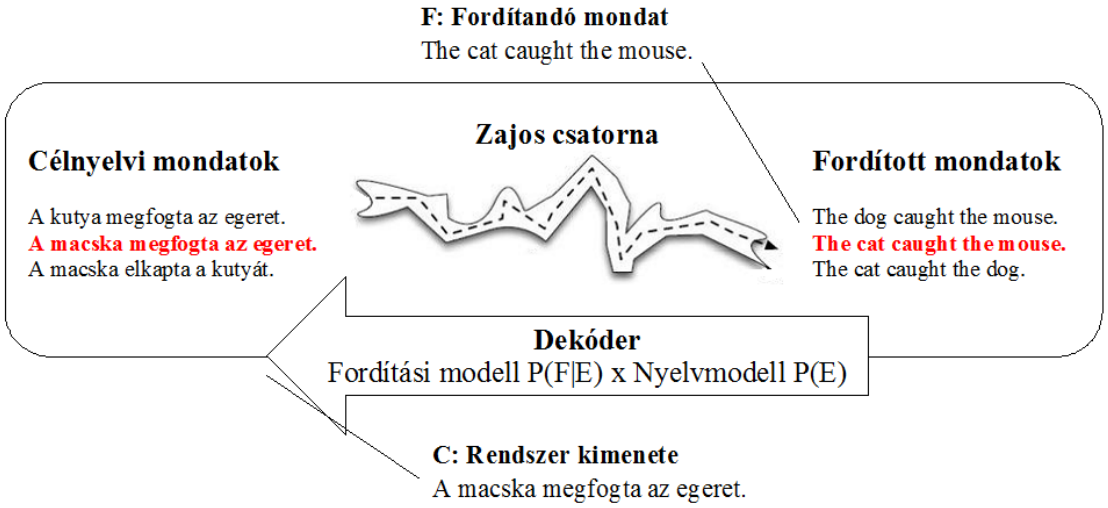
2.2. ábra Fordítómemória szerepe példaalapú gépi fordításban

2.1.3. Statisztikai gépi fordítás

A statisztikai gépi fordítás (Statistical Machine Translation - SMT) [16] párhuzamos szövegkorpuszon alapszik. A példaalapú gépi fordítás kialakulásával egyidejűleg alakult ki. Hasonlóan a példaalapú módszerhez, ez is az emberek által előre lefordított szövegek felhasználásán alapszik. A párhuzamos korpuszok előre lefordított nagy mennyiségű szövegek. A rendszer a párhuzamos korpuszt elemezve felállít egy szótárat, illetve statisztikai módszereken alapuló megfigyeléseket tesz, amelyek alapján fordít.

A probléma megoldásához a beszédtechnológiában használt Shannon-féle zajoscsatorna-modellt vették alapul [18]. A rendszer modelljét (lásd 2.3. ábra) az alábbi módon írhatjuk le. Adott egy F forrásnyelv, amit szeretnénk lefordítani E célnyelvre. Amit biztosan ismerünk, az a lefordítandó szövegünk. A fordítás úgy történik, hogy vesszük az összes lehetséges célnyelvi mondatot, amelyek mintha a forrásnyelvi mondat zajos verziói lennének. Majd ezeket a „zajos mondatokat" összehasonlítjuk a fordítandó mondatunkkal. Az lesz a rendszer kimenete (C), amelyik „zajos mondat" a legjobban hasonlít rá, vagyis amelyik a legnagyobb valószínűséggel a fordítás (Bayes-féle becsléssel számítjuk ki a P valószínűséget).

DOI:10.15774/PPKE.ITK.2019.003



2.3. ábra Zajos csatorna modell

A statisztikai gépi fordítórendszer felépítéséhez szükség van egy fordítási modellre és egy nyelvmodellre, valamint egy dekódoló rendszerre, amely összeköti a fordítási modellt a nyelvmodellel, és megtalálja a legvalószínűbb fordítást, a forrásnyelvi mondat alapján. A fordítási modell felel a fordítás tartalomhűségéért (adequacy), míg a nyelvmodell felel a célnyelvi mondat gördülékenységéért és nyelvhelyességéért (fluency).

A fordítási modellben a fordítási egységek – amelyek egyben meghatározzák a statisztikai gépi fordítórendszer fajtáját is – lehetnek szavak (szóalapú statisztikai gépi fordítórendszer), kifejezések (kifejezésalapú statisztikai gépi fordítórendszer), tulajdonsághalmazok (faktoralapú statisztikai gépi fordítórendszer) és generatív szabályok (szintaxisalapú vagy hierarchikus statisztikai gépi fordítórendszer). Ezekből a két legnépszerűbb módszer a kifejezésalapú és a hierarchikus.

A kifejezésalapú statisztikai gépi fordítórendszer [19] a statisztikai gépi fordítás egyik fajtája, amelynek fordítási modellje kifejezések összekötésein alapszik. Az összekötött kifejezéseket egy párhuzamos korpusz segítségével állítja elő, automatikus statisztikai módszerekkel. Azoknál a nyelveknél, melyek szintaktikailag, illetve szórendileg hasonlóak egymáshoz, a rendszer nagy pontossággal tud fordítani. A kifejezésalapú módszer lokális átrendezésekkel tudja javítani a fordítás minőségét, de nagyobb távolságokat nem tud kezelni. Ezért azoknál a nyelveknél, ahol nagyobb a különbség szórendileg, kisebb pontossággal fordít a rendszer.

DOI:10.15774/PPKE.ITK.2019.003

A hierarchikus statisztikai gépi fordítórendszer [20] ezt a problémát igyekszik megoldani: képes nagyobb távolságú átrendezéseket végezni. A hierarchikus módszer a kijefezésalapú módszer bővített változata. Amíg a kijefezésalapú módszer kifejezésalapú dekódert használ, addig a hierarchikus módszer környezetfüggő nyelvtant használó dekódert. Ez a módszer komplexebb átrendezési szabályokat segít megtanulni a hierarchikus rendszernek. Például: az angol-francia tagadáspárost *don't* $X \rightarrow$ *ne* X *pas* formába menti el, ahol az X helyére bármilyen igei szerkezet behelyettesíthető.

A statisztikai gépi fordítás legnagyobb előnye, hogy nem kell ismerni az adott nyelvpárt, amelyekre alkalmazni szeretnénk a rendszert; bármilyen két nyelvet be lehet tanítani. Továbbá nincsen szükség emberi közreműködésre. Ráadásul az internet széleskörű elterjedésének köszönhetően igen nagy mennyiségű többnyelvű digitális szöveg áll rendelkezésre. Hátránya, hogy ha kevés a tanítóanyag, akkor nem lesz jó a modell minősége. Ha pedig a tanítóanyag minősége rossz, akkor a betanított modell sem lesz jó. Tehát nagyban függ a modell hatékonysága a korpusz minőségétől. Továbbá, ha téma (domain) specifikusan tanítjuk be a modellt, abban az esetben a gépi fordítórendszer csak az adott témán belül képes megfelelő minőségben fordítani.

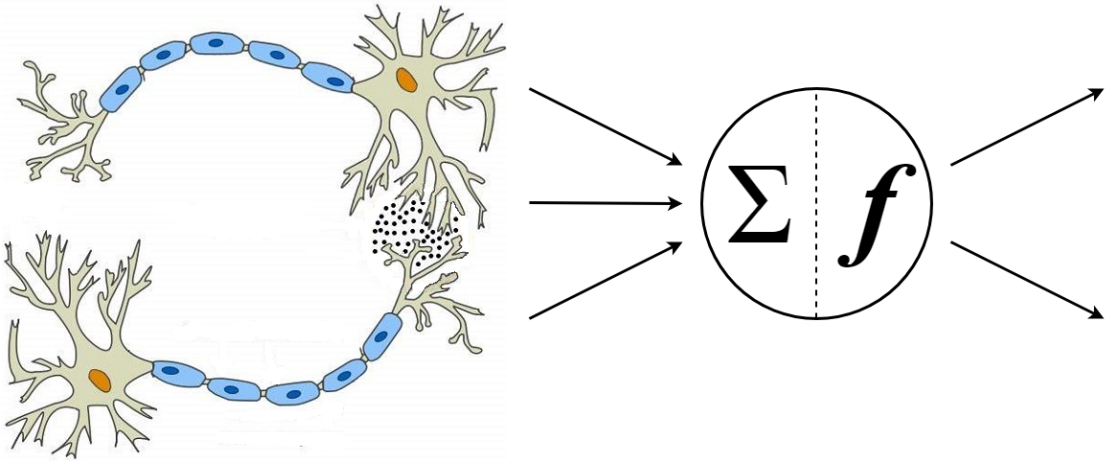
2.1.4. Hibrid gépi fordítás

A tényleges alkalmazások során, a jobb eredmény érdekében gyakran ötvözik a különböző gépi fordító módszerek előnyeit. A hibrid gépi fordítás (Hybrid Machine Translation) célja, hogy egy adott típusú gépi fordító módszer minőségét és pontosságát javítsa más gépi fordító módszerek integrálásával [21–23]. Ilyen például, ha a szabályalapú döntéseknél a rendszer figyelembe vesz statisztikai eredményeket, vagy amikor statisztikai fordításnál szabályrendszert is alkalmaznak. További lehetőség, hogy a pontosabb fordítás érdekében, mind a szabály, mind a statisztikai módszernél fordítómemóriát alkalmaznak.

2.1.5. Neurális gépi fordítás

A neurális hálózat elméletét már 1943-ban megfogalmazták [24]. A neurális gépi fordítás alapja a mesterséges neurális hálózat, amely mesterséges neuronokból épül fel. A mesterséges neuronok az idegsejteket modellezik. Az emberi agy több milliárd idegsejtet

DOI:10.15774/PPKE.ITK.2019.003



2.4. ábra Neuron modellezése [25]

tartalmaz, ezek működésének hatékonysága ihlette a mesterséges neuronok létrehozását. A neuron egy információ-feldolgozó egység, amely bemeneti adatokból számításokkal generál kimeneti adatot (lásd 2.4. ábra).

A számítás első része lehet a bemeneti értékek egyszerű súlyozott összegzése ($\sum$), de lehet komplexebb művelet is. A számítás második része – ami esetenként elmaradhat – egy átviteli függvény (f), ami a súlyozott bemenetekből állítja elő a kimeneti értékeket. Az átviteli függvény általában egy küszöb vagy egy szigmoid-függvény, de ez is lehet komplexebb művelet (pl.: Gauss-féle függvény).

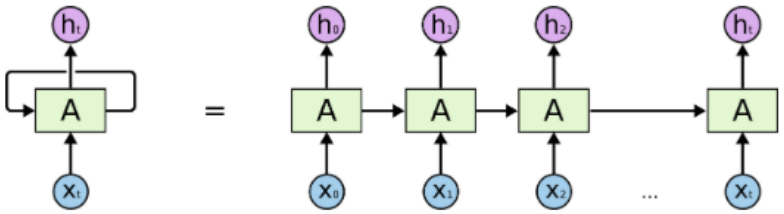
A mesterséges neurális hálózat a mesterséges neuronok összekapcsolásával jön létre. A neurális hálózat három, funkcionálisan és strukturálisan elkülöníthető rétegre osztható:

1. Bemeneti réteg: a hálózat a bemeneti rétegen keresztül kapja az információt a külvilágtól, amit több réteg esetén módosítás nélkül továbbít a hálózat további részébe.

2. Kimeneti réteg: a kimeneti réteg állítja elő a hálózat eredményét. Feladattól függően változik a kimeneti réteg mérete. Például osztályozási feladat esetében a kimeneti rétegben annyi neuron van, ahány osztály.

3. Rejtett rétegek: a rejtett rétegek a bemeneti és a kimeneti rétegek között helyezkednek el. A legegyszerűbb neurális hálózat egy rejtett réteggel rendelkezik. Minél több rejtett rétege van a hálózatnak, annál jobban növekszik az absztrakciós képessége és annál összetettebb feladatokat képes megoldani. A rejtett rétegek számának

DOI:10.15774/PPKE.ITK.2019.003



2.5. ábra A rekurrens neurális hálózat működése [27]

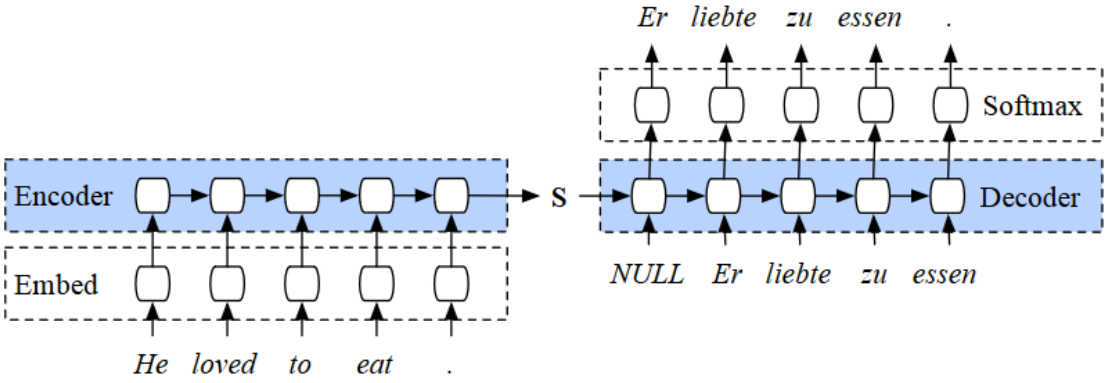
növelésével „mélyül” a hálózat, ezért a „mély neuronhálózat” (deep neural network) kifejezés a rétegek számára utal, a „mély tanulás” (deep learning) kifejezés pedig a több rejtett réteggel rendelkező neurális hálózat tanítását jelenti.

A neurális hálózat topológiája alapján lehet előrecsatolt és visszacsatolt. A visszacsatolt hálózat esetében nem csak előre csatolások, hanem a rétegeken belül és a rétegek között is lehetnek visszacsatolások.

A neurális hálózat módszer egyik legnagyobb előnye, hogy képes tanulni a saját hibájából. A hálózat a rendszer hibáját egy veszteségfüggvénnyel és várt kimenetek segítségével számolja ki. A veszteségfüggvény lehet átlagos négyzetes eltérés (mean squared error), különböző kereszt-entrópia (cross-entropy) függvények stb. A hiba meghatározásánál először a kimeneti hibákat számolja ki, majd a hiba-visszaterjesztés alkalmazásával kiszámolja a rejtett rétegek hibáit is. Amikor a rendszer a hibákat kiszámolta, az egyes rejtett rétegekben a súlyokat úgy állítja át, hogy az adott rétegre számolt hibamérték csökkenjen. Ezt a hiba-visszaterjesztési folyamatot a tanítás során többször elvégezi, optimális esetben addig, amíg a hibamérték minimális nem lesz.

A neurális gépi fordítás (Neural Machine Translation - NMT) [26] rekurrens neurális hálózatot (Recurrent Neural Network – RNN) használ. Az RNN szekvenciális bemneti adatra kitalált módszer. Ahogy a 2.5. ábra mutatja, az RNN egy visszacsatolt hálózat, amely egy bemeneti szöveg egységein iterálva kiszámol egy súlyozott értéket, ami az adott bemeneti szöveget fogja jellemezni. Az ábrán az x a bemenet, a h a kimenet, és az A egy rejtett rétegben lévő neuron. A szöveg egysége lehet szó, szórészlet vagy akár betű (karakter) is.

DOI:10.15774/PPKE.ITK.2019.003



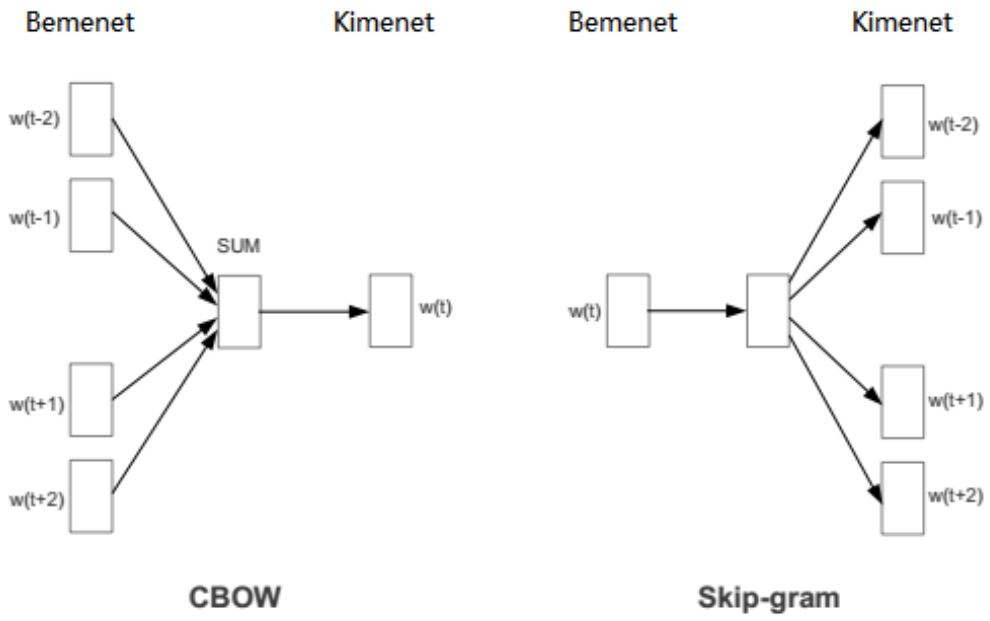
2.6. ábra A neurális gépi fordítás működése [31]

A neurális gépi fordítórendszereket [28] leggyakrabban egy „enkóder-dekóder” (encoder-decoder) architektúrájú neurális hálózat segítségével állítják elő [28–30], amely kettő RNN-hálózatból épül fel (lásd 2.6. ábra kékkel jelölt részei). Az enkóder felel azért, hogy a változó hosszúságú bemeneti szövegből leképezzen egy állandó hosszúságú vektor reprezentációt (S), amivel majd a dekóder dolgozik. A dekóder feladata az, hogy az enkóder által adott fix hosszúságú vektor reprezentációból kimeneti szöveget generáljon.

A 2.6. ábrán látható egy-egy plusz réteg, mind a bemenetnél (Embed), mind a kimenetnél (Softmax). Ezek a rétegek az úgynevezett szóreprezentáció rétegek. A szóreprezentáció lényege, hogy az RNN nem közvetlenül a nyers bemeneti szöveget dolgozza fel, hanem a szöveg egységeiből (jelen esetben szavak) szóvektorokat készít, és ezt adja oda az RNN-nek. A szóvektor előállítása történhet az úgynevezett „one-hot” vektor módszerrel. A „one-hot” vektor egy olyan vektor, amiben csak 0-ák állnak, kivéve egyetlen elemet, ami egy 1-es. A vektor hossza egyenlő a neurális gépi fordítórendszer tanításához használt tanítóanyagból létrehozott szótár hosszával. A vektorban szereplő 1-es indexe megegyezik az általa reprezentált szó szótárbeli indexével (helyének sorszáma). Később, a „one-hot” vektor módszere mellett, a szóbeágyazás (Word Embedding - WE) módszere lett a legnépszerűbb szóreprezentációs módszer, ami szemantikai információt tartalmaz.

A szóbeágyazás [32, 33] azon az elméleten alapszik, hogy a hasonló jelentésű szavakat hasonló környezetben használjuk. A szóbeágyazás módszerében a lexikai elemek egy valós vektortérben egy-egy pontnak felelnek meg, amelyek konzisztensen helyezkednek el az adott térben. Ebben az adott térben a szemantikailag közel álló szavak közel esnek egymáshoz, míg a jelentésben távol álló szavak távol vannak egymástól. A szemantikai

DOI:10.15774/PPKE.ITK.2019.003



2.7. ábra CBOW és Skip-gram működése [33]

hasonlóságot a két pont közötti távolsággal írhatjuk le. A modell tanításához újabb neurális hálózatra van szükségünk. Kétféleképpen taníthatjuk a modellt. Az első eset, amikor a neurális hálózat bemenetei egy szó, és a kimenete a bementi szó fix méretű környezetének szavai. Ezt a Skip-gram modell (lásd 2.7. ábra) segítségével tudjuk betanítani. A másik lehetőség, hogy a hálózat bemenete egy szó fix környezetének szavai, és a kimenet az adott szó, aminek a környezetét vizsgáljuk. Ezt a CBOW (Continous Bag-of-Words) modell (lásd 2.7. ábra) segítségével lehet betanítani.

A módszer előnye az SMT-vel szemben, hogy képes tanulni a saját hibáiból, sokkal gördülékenyebb fordításokat hoz létre. Hátránya viszont, hogy ezt képes a tartalmi pontosság rovására tenni.

A GPU (Graphics Processing Unit) technológia dinamikus fejlődésének, valamint megfizethető árú videokártyáknak köszönhetően, a mélytanulás-alapú rendszerek elérhetővé váltak a kutatók számára. A neurális hálózatokon alapuló rendszerek a legtöbb tudományterületen legyőzték teljesítményben az addig legjobbnak számító rendszereket. A gépi fordítás területen is átvették a neurális módszerek a kutatások fókuszát.

DOI:10.15774/PPKE.ITK.2019.003

2.2. Gépi tanulás

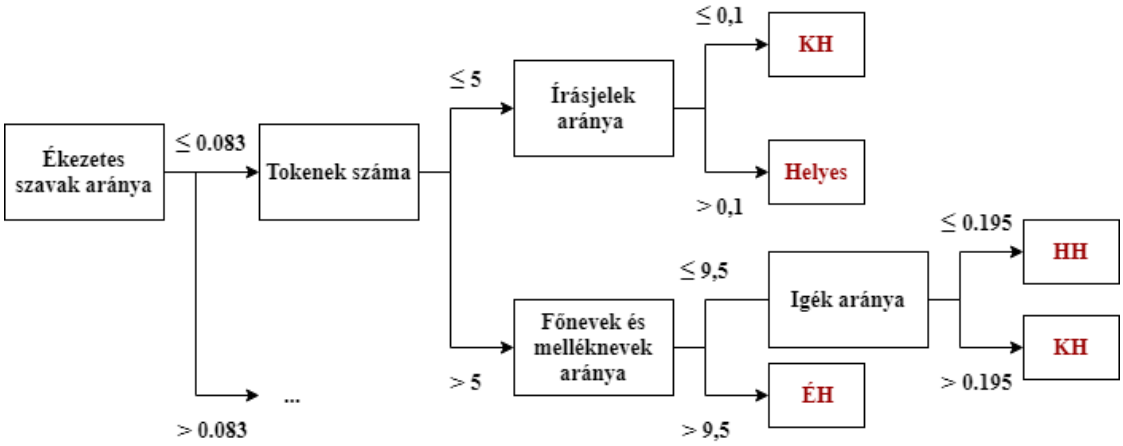
A minőségbecslés módszere gépi tanulás (Machine Learning) módszerein alapszik, mint az osztályozás vagy a regresszió. A gépi tanulás képes adott adatokból tanulni, és a megtanult tapasztalatokból, mintákból predikciót végezni, döntést hozni vagy tudást generálni. A gépi tanulás lehet felügyelt és felügyelet nélküli. A felügyelt gépi tanulás megadott bemeneti példák és azoknak elvárt kimeneti eredményeiből tanul. Célja olyan szabályok megfogalmazása, amelyekkel létre tud hozni egy olyan leképezési modellt, amely a bemeneti adatokat összeköti a kimeneti eredményekkel. A felügyelet nélküli gépi tanulás esetében nem áll rendelkezésünkre elvárt kimeneti eredmény. A módszer célja, hogy a bemeneti adatokban valamilyen mintázatot, karakterisztikát találjon.

A minőségbecslés módszere gépi tanuláson alapszik, ezért a következő alfejezetekben röviden bemutatom a fontosabb gépi tanulási módszereket.

2.2.1. Döntési fa és véletlen erdő

A döntési fa (Decision Tree) [34] egy felügyelt gépi tanulási módszer, amely osztályozásra alkalmas. Egy betanított modell esetén a predikció úgy történik, hogy a modell bemenetként egy attribútumokkal rendelkező objektumot vagy szituációt kap, majd egy tesztsorozat révén a bemenet attribútumára vonatkozóan kérdések sorozatát teszi fel. A módszer a kérdésekre a bemeneti adatokból kap válaszokat, amelyek alapján jut el a következtetésre, a kimenethez. A kimenet, az osztályattribútum egy nominális attribútum. A kérdések és a válaszok sorozata fa struktúrába rendezhető, amely egy hierarchikus struktúrát ír le. A fa csúcsokból és irányított élekből áll. Pontosan egy gyökér csúccsal rendelkezik: itt kezdődik a bemeneti objektum feldolgozása, majd a gyökér csúcsból a belső csúcsokon keresztül jut el valamelyik levélig, ami a rendszer kimenetét adja, amin az osztályattribútum egy-egy értéke szerepel. A gyökér csúcsnak nincsen bemeneti éle, és nulla vagy több kimeneti éle van. A köztes csúcsoknak egy bemeneti éle van, és egy vagy több kimeneti éle. A leveleknek egy bemeneti éle van, és nincsen kimeneti éle. A 2.8. ábrán látható egy konkrét példa, ahol az osztályattribútum egy szöveg hibáinak lehetséges típusait jelöli: KH a központozás hibái (hiánya), nagybetűk elhagyása; HH az elírások, helyesírási és nyelvi hibák; és $ÉH$ az ékezetek hiánya.

DOI:10.15774/PPKE.ITK.2019.003



2.8. ábra Példa döntési fára

A döntési fa tanítása során rekurzívan állítjuk elő magát a fát. A megadott tanítóanyagból kiindulva olyan kérdéseket keresünk, amelyek segítségével részeire tudjuk bontani a tanulóhalmazt. A cél, hogy minél kisebb mélységű legyen a fa. Ennek eléréséhez midnen lépésben azt az attributumot választjuk, amelynek segítségével a legnagyobb bizonysággal tudja elvégezni a predikciót.

Egy bontást (vagy szétvágást) akkor tekintünk jónak, ha a magyarázandó változó eloszlása a szétvágott halmazokban kevésbé szórt, mint a vágás előtt. A keletkező részekre rekurzívan alkalmazzuk a szétvágás műveletét, amíg van attribútum, ami alapján oszthatjuk az elemeket, vagy amíg van olyan bontás, amely javítani tud az aktuális osztályon. Ha beállítottunk egy mélységi korlátot a fának, akkor az adott mélység elérésével is megáll a tanítás. Amikor elérjük a levél szintjét, minden levélhez hozzárendelünk egy döntést.

A véletlen erdő alapötlete [34], hogy sok döntési fát használunk. Mindegyik döntési fa különbözik egymástól. Az osztályozás során mindegyik döntési fa ad egy predikciót, melynek összegzése szavazással történik. Amelyik válasz a legtöbb szavazatot kapta az lesz a végső döntés eredménye. A véletlen erdő hatékonysága függ a döntési fák számosságától és a döntési fák közötti korreláció mértékétől.

DOI:10.15774/PPKE.ITK.2019.003

2.2.2. Lineáris regresszió

Az osztályozás esetében a tanult függvény értékkészlete diszkrét. Amennyiben folytonos az értékkészlet, regresszióról beszélünk.

A lineáris regresszió [34] a magyarázóváltozók (X) és a magyarázott (y) változó között keres és feltételez lineáris kapcsolatot, vagyis az y jó közelítéssel az X_i változók lineáris függvényeként áll elő. Adott n darab (magyarázóváltozók száma) mintaanyag, amelyek pontfelhőt alkotnak. Feladatunk erre a pontfelhőre ráilleszteni egy egyenest. Ennek az egyenesnek a segítségével meg tudjuk becsülni y változását az X változók változásának függvényében. A lineáris kapcsolat y és X között az alábbi függvénnyel fejezhető ki:

$$y = \beta X + u = \beta_0 + \beta_1 x_1 + \beta_2 x_3 + ...\beta_n x_n + u.$$

A lineáris regresszió feladata a β paramétervektor becslése. A magyarázóváltozók számától függően lehet egyszerű és többszörös lineáris regresszió. A lineáris regresszió tanítása során a megadott mintákból a β paramétervektort számítjuk ki, valamilyen becslési módszerrel. A legegyszerűbb becslési módszer a legkisebb négyzetek módszere.

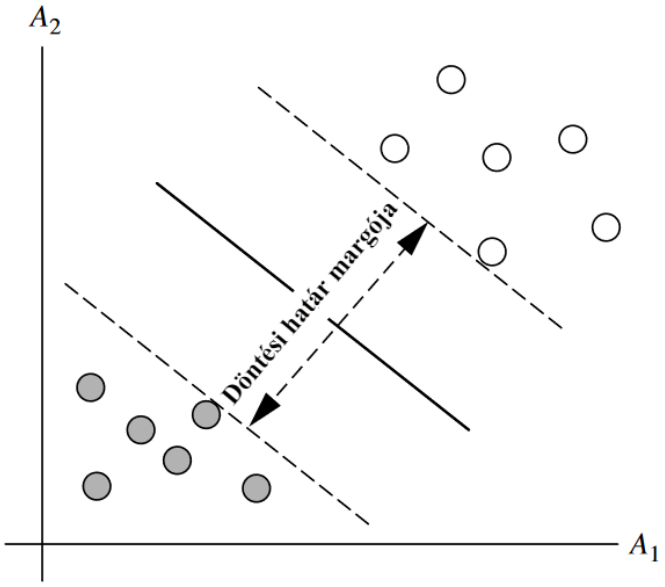
2.2.3. Szupport vektor gépek és szupport vektor regresszió

A szupport vektor gépek (Support Vector Machines - SVM) [34] egy gépi tanulási módszer, amely osztályozásra és regresszió analízisre alkalmas.

Adott egy tanítóhalmaz, mintaadatokkal, és két osztály. Minden mintaadat egyik vagy másik osztályba tartozik. A SVM egy lineáris osztályozó modell segítségével próbálja besorolni az új adatot egyik, vagy másik osztályba. Ha a mintaadatunkat egy térbeli ponttal reprezentáljuk egy d dimenziójú vektortérben, akkor az SVM a mintapontokat egy $d-1$ dimenziójú hipersíkkal osztja két osztályba. Mivel egy ilyen osztályozási feladatra több hipersík is alkalmas lehet, ezért az SVM igyekszik a legoptimálisabb megoldást megtalálni, ami nem más, mint a modell általánosító-képességének maximalizálása. Az SVM úgy oldja meg ezt a problémát, hogy bevezeti a maximális margó fogalmát. Ha kétdimenziós térben vagyunk, akkor a margó, az az osztályozó döntés határának (elválasztó

DOI:10.15774/PPKE.ITK.2019.003

hipersík) két oldalán lévő, egyenlő távolságú két párhuzamos egyenessel meghatározott térrésze (lásd 2.9. ábra). A margó célja, hogy a modell általánosító-képességét növelje, ezért a margó nem tartalmaz mintapontot, és emellett mérete maximális.



2.9. ábra Szupport vektor gépek margója [34]

Vannak nem szeparálható esetek. Ilyenkor úgy határozzuk meg a maximális margójú elválasztó hipersíkot, hogy minimális hibát megengedünk a rendszernek.

Az SVM képes nemlineáris összefüggések tanulására is [35]. Ezt kernel függvény segítségével oldja meg. A kernel gépek lényege, hogy amikor egy regressziós feladatban a mintapontok lineárisan nem szeparálhatóak, akkor egy nemlineáris leképezéssel a bementi térből egy úgynevezett jellemzőtérbe képezi le őket. Ezt követően egy transzformációval egy kernel reprezentációba tér át, amely a jellemzőtérbeli reprezentációból belső szorzattal kiszámolható. A jellemzőtérbeli leképezés célja, hogy az adott mintaadatokat leképezi egy magasabb dimenzióba, majd az új vektortérben lineárisan szeparálja őket, vagyis egy nemlineáris transzformációval lineárisan szeparálhatóvá alakítja a feladatot. Az SVM módszerét ki lehet terjeszteni regressziós feladatokra is, ez a szupport vektor regresszió (Support Vector Regression - SVR).

DOI:10.15774/PPKE.ITK.2019.003

2.2.4. Gauss-folyamat

A Gauss-folyamat [36, 37] alkalmazható regressziós feladatok megoldására. A Bayes-becslés és a kernel gépek együttműködésén alapszik.

A módszer alapja a Naive Bayes-becslés [34], amely az attributumokhoz valószínűségi változókat rendel, és az osztályattributum értékét – amire tanítottuk az osztályozót – a valószínűségi változók többi változóra vett feltételes eloszlása alapján becsli amelynek alapja a Bayes-tétel:

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}$$

, ahol a $P(X)$ és $P(Y)$ megfigyelt események valószínűségei, a $P(Y|X)$ feltételes valószínűség, ami azon megfigyelésen alapszik, hogy X bekövetkezésekor Y is bekövetkezett. A Bayes-féle módszerből kiindulva regresszióra is alkalmas a Gauss-eljárás. A lineáris regresszió módszerét módosítja azzal, hogy feltételes Gauss-eloszlást számít a pontok becslése helyett. Hasonlóan, a szupport vektor regresszió módszerhez, kernel gép segítségével oldja meg a lineárisan nem szeparálható feladatokat.

2.2.5. Együttes módszerek

A zsákolás (bagging) és a gyorsítás (boosting) módszereket együttes (ensenmble) módszereknek hívják. Ezek különböző gépi tanuló algoritmusokat kombinálva érnek el jobb eredményt [34]. Ezen módszerek lényege, hogy különböző gépi tanuló módszerek predikciói között tartanak szavazást, és amelyik kimeneti érték a legtöbb szavazatot kapta, az lesz a rendszer végső kimenete. Alapötlete hasonlít a véletlen erdő módszerére, csak itt nem döntési fákat, hanem különböző osztályozó módszerket egyesít.

A zsákolást bootstrap aggregálásnak (bootstrap aggregating) is hívják. A módszer alapja, hogy a tanítóanyagból egyenletes eloszlással, véletlen mintavételezéssel, vele azonos méretű bootstrap mintákat hoz létre, majd a bootstrap mintahalmazokra hoz létre osztályozókat. Végül a bootstrap mintahalmazokra adott predikciók alapján hozza meg a végső döntést. A zsákolás abban az esetben működik jól, ha a kombinált modellek működésükben különbözőek.

DOI:10.15774/PPKE.ITK.2019.003

A gyorsítás egy iteratív módszer. Abban különbözik a zsákolástól, hogy a mintahalmazokat egymástól függően hozza létre. A módszer minden mintahalmaz létrehozásakor figyelembe veszi az előző lépésben létrehozott mintahalmazon mért eredményeket. Az algoritmus nagyobb súlyokat rendel a nehezen osztályozott esetekhez, amelyek ezáltal nagyobb eséllyel kerülnek be a következő minthalmazba. Így a nehezen osztályozható esetekre több figyelmet szentel.

2.2.6. Jegykiválsztás

A gépi tanulás egyik legfontosabb feladata a jegykiválsztás (feature selection), azaz a releváns jegyhalmaz megtalálása. A jegykiválasztással azokat a jegyeket keressük meg, amelyek a legnagyobb hatással vannak a predikcióra nézve. Egy gépi tanulás feladatban akár több száz jegy is előfordulhat, de nem mindegyik jegy lesz alkalmazható az adott feladatra, sőt lehetnek közöttük olyan jegyek is, amelyek rontják a modell teljesítményét. Egy másik fontos szempont, hogy a releváns jegyek kiválasztásával csökkentjük a bemeneti jegyek terének dimenzióját is, ami egyben a program erőforrásigényének optimalizálását is jelenti.

Az egyik megközelítés a korreláció alapú jegykiválasztás [38] (Correlation-based Feature Selection - CFS). A módszer lényege, hogy megkeresi azokat a jegyeket, amelyek magasan korrelálnak a kimeneti értékekkel, de egyúttal a jegyek egymás között alacsonyan korrelálnak. A módszer kiválasztja a legerősebb befolyású jegyeket, miközben kizárja a redundáns jegyeket.

Egy másik népszerű jegykiválasztási módszer a döntési fák által nyújtott attribútumsúlyok alkalmazása. A döntési fák egyik tulajdonsága, hogy rangsorolják a jegyeket asszerint, hogy azok mennyire jellemzik a kimenetet. A rangsor alapján készítik el a kérdések sorozatát, vagyis építik fel a fát. Ezt a tulajdonságot felhasználva ki tudjuk nyerni a releváns jegyhalmazt.

Egy lassabb, de pontosabb megoldást nyújt a „forward selection” módszere. Kezdetben a jegyhalmazunk üres. Az első lépésben megvizsgáljuk, hogy melyik jegy van a legnagyobb hatással a kimenetre: ezt bevesszük a jegyhalmazba. Majd következő lépésben

DOI:10.15774/PPKE.ITK.2019.003

megvizsgáljuk, hogy melyik másik jegy hozzáadásával tudjuk elérni a jobb eredményt: ezt pedig hozzáadjuk a jegyhalmazhoz. Ezt addig ismételjük, amíg eredményjavulást tudunk elérni.

2.3. A WordNet

A WordNet [39] egy nyelvi ontológia. Az ontológia [40] a mesterséges intelligencia területén, a tudás reprezentálására alkalmas. Célja a világ lényegi dolgainak ábrázolása és az általa reprezentált tudáshalmaz megosztása és újrafelhasználása. Tudásbázisnak is szokás nevezni.

A WordNet egy speciális lexikális szemantikai hálózat. A hálózat csomópontjai a szinonimahalmazok (synset). A szinonimahalmazok azonos jelentésű fogalmakból, szinonimákból állnak. Egy konkrét példa: {Canis familiaris, házikutya, kutya, eb}. A hálózat csak a tartalmi szófajokat tartalmazza: főnév, ige, melléknév és határozószó.

A WordNet csomópontjai közötti élek a szinonimahalmazok közötti szemantikai relációkat jelentik. A relációkat pszicholingvisztikai kutatások motiválták. A főnév esetében a legfontosabb reláció a *hipernima* (ellentétje: *hiponima*), ami hierarchikus alá-/fölérendeltséget vagy specifikus/generikus viszonyt fejez ki. Például a {Canis familiaris, házikutya, kutya, eb} hipernimája a {háziállat, háziasított állat}. Hasonló reláció a *meronima* (ellentétje: *holonima*), ami rész-egész viszonyt fejez ki (például: {fa} - {erdő}). Az igék esetében is a legfontosabb reláció a hipernima (ellentétje: *troponima*), ami egy hierarchikus kapcsolatot fejez ki (például: {élőlény létezik} - {életben van, él}). A melléknevek esetében két fontos reláció az *antonima* és a *similar_to*. Az *antonima* egy tágabb értelembe vett ellentétet fejez ki (például: {jó} - {rossz}), a *similar_to* pedig két fogalom közötti hasonlóságot (például: {jó} - {megfelelő}). A határozószók esetében szintén két fontos reláció az *antonima* (például: {lassan, megfontoltan} - {gyorsan, sebesen}) és a *eq_near_synonym*. Az utóbbi, két fogalom közötti hasonlóságot fejez ki (például: {körültekintően, gondosan, megfontoltan} - {gondosan, figyelmesen}).

DOI:10.15774/PPKE.ITK.2019.003

A Magyar WordNet [39] több mint 42 ezer szinonimahalmazt tartalmaz: mintegy 33500 főnévi, 3600 igei, 4000 melléknévi és 1000 határozószói. Továbbá a szinonimahalmazok egy része tartalmaz angol azonosítót is, amely a Princeton WordNet 3.0 [41] szinonimahalmazainak azonosítója. Így a két WordNet között egyértelmű leképezést végezhetünk.

2.4. Néhány fontos metrika

Az alábbiakban röviden bemutatok néhány metrikát, amelyeket felhasználtam kutatásom során.

2.4.1. Pontosság, Fedés, F-mérték

Bináris osztályozási feladathoz gyakran használt kiértékelési módszer a pontosság, a fedés és az F-mérték [42]:

- Pontosság (Precision): Azt méri, hogy milyen arányban állnak a helyesen kiértékelt elemek az eredményhalmazban lévő összes elemmel. Nem azonos a kiértékelés szempontjaiban tárgyalt pontossággal (adequacy) (lásd 3.2. fejezet).

$$pontosság = \frac{helyesen\ kiértékelt\ elemek\ száma}{eredményhalmaz\ elemeinek\ száma}$$

- Pontosság a gépi fordításban:

$$pontosság = \frac{helyesen\ fordított\ szavak\ száma}{gépi\ fordítás\ szavainak\ száma}$$

- Fedés (Recall): Azt méri, hogy milyen arányban áll a helyesen kiválasztott elemek száma a célhalmazban lévő összes kiválasztandó elem számával.

$$fedés = \frac{helyesen\ kiértékelt\ elemek\ száma}{célhalmaz\ elemeinek\ száma}$$

- Fedés a gépi fordításban:

$$fedés = \frac{helyesen\ fordított\ szavak\ száma}{referenciafordítás\ szavainak\ száma}$$

DOI:10.15774/PPKE.ITK.2019.003

- F-mérték (F-score, F1 score): A pontosság és a fedés súlyozott mértani átlaga.

$$F\text{-}mérték = 2 * \frac{pontosság * fedés}{pontosság + fedés}$$

2.5. A minőségbecslés teljesítményének mérése

A minőségbecslő rendszer kiértékeléséhez az átlagos abszolút eltérés (Mean Absulute Error – MAE), az átlagos négyzetes eltérés gyöke (Root Mean Squared Error – RMSE), a Pearson-féle korreláció és a helyesen osztályozott egyedek (Corrected Classified Instances - CCI) metrikákat használtam [34].

Az átlagos abszolút eltéréssel két folytonos változó közötti különbséget lehet kiszámolni. Képlete:

$$MAE = \frac{\sum_{i=1}^{n} |X(s_i) - Y(s_i)|}{n}$$

, ahol X a teszthalmazban lévő referenciakiértékelések, Y a teszthalmazban lévő gép által becsült értékek, s_i az i-dik eleme a teszthalmaznak, és n a teszthalmaz mérete. A képlet az átlagtól való eltérést számolja ki. Minél kisebb az eltérés az átlagtól, annál kevésbé variábilis a rendszer, vagyis annál inkább hasonlóak a becsült értékek a referenciaértékekhez.

Az átlagos négyzetes eltérés gyöke hasonló az átlagos abszolút eltérés módszeréhez, annyi különbséggel, hogy sokkal érzékenyebb a kiugró értékekre:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (X(s_i) - Y(s_i))^2}{n}}$$

A Pearson-féle korreláció két érték közötti lineáris kapcsolat mértékét mondja meg. Ha a korreláció 1, vagy ahhoz közeli, akkor a két érték erősen összefügg egymással és hasonló a viselkedésük. Ha 0, vagy ahhoz közeli érték, akkor függetlenek egymástól, vagy gyengén függnek egymástól. Ha -1, vagy ahhoz közeli, akkor szintén erősen összefüggnek, csak ellentétes irányú a viselkedésük. A korrelációt a korrelációs együtthatóval (r) számítjuk ki:

$$r = \frac{\sum_{i=n}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=n}^{n} (x_i - \bar{x})^2} \sqrt{\sum_{i=n}^{n} (y_i - \bar{y})^2}}$$

DOI:10.15774/PPKE.ITK.2019.003

, ahol x_i az i-dik becsült érték, y_i az i-dik referenciaérték, n a teszthalmaz mérete, az $\bar{x} = \frac{1}{n}\sum_{i=i}^{n} x_i$ (a becsült értékek átlaga) és $\bar{y} = \frac{1}{n}\sum_{i=i}^{n} y_i$ (a referenciaértékek átlaga).

A *helyesen osztályozott egyedek* metrika pontosságot számol osztályozós feladatokra:

$$CCI = \frac{helyesen\ osztályozott\ egyedek\ száma}{teszthalmaz\ mérete}$$

2.5.1. Látens szemantikai analízis

A látens szemantikai analízis (Latent Semantic Analysis - LSA) [43] módszerével szövegek közötti szemantikai hasonlóságot tudunk mérni. A módszer alapja a vektortérmodell, ami a dokumentumokat – jelen esetben a mondatokat – egy sokdimenziós vektortérben ábrázolja. A vektortérmodell dimenzióit a felhasznált korpuszból előállított szótár szavai alkotják. A dokumentumokat egy szó-dokumentum mátrixban ábrázolja, ahol a sorok a szótár szavaiból állnak, az oszlopok pedig a dokumentumokból (lásd 2.1. táblázat). A mátrix sorainak száma egyenlő a szótár méretével, az oszlopok száma megegyezik a dokumentumok számával. A mátrixban egy cella értéke 0, ha a cellához tartozó szó nem szerepel a cellához tartozó dokumentumban és 1 a cella értéke, ha szerepel.

	S1	S2	S3	S4
romeo	1	0	1	0
juliet	1	1	0	0
happy	0	1	0	0
dagger	0	1	1	0
live	0	0	0	1

S1: Romeo and Juliet.
S2: Juliet: O happy dagger!
S3: Romeo died by dagger.
S4: „Live free or die”.

2.1. táblázat Példa az LSA-ra

A mátrixban az 1-es értékeket a *tf-idf* értékkel súlyozza. Ez a *tf* (Term Frequency) és az *idf* (Inverse Document Frequency) érték szorzata, ahol a *tf* azt mondja meg, hogy az adott t szó hányszor szerepel az adott dokumentumban, és *idf* azt mondja meg, hogy az adott t szó hány dokumentumban fordul elő – vagyis mennyire informatív az adott szó a dokumentumokra nézve. Így egy adott szót a hozzá tartozó mátrixban lévő sorvektor reprezentálja.

DOI:10.15774/PPKE.ITK.2019.003

Az LSA egy új mondat vizsgálatakor a szavakból egy szózsákot hoz létre és a szó-dokumentum mátrix segítségével, szinguláris értékfelbontással a szózsákból egy látens szemantikai index vektort számol ki (Latent Semantic Indexing - LSI). Két mondat LSI vektor közötti hasonlóságának értéke a két vektor koszinusz távolsága.

Kétnyelvű LSI esetén a szótár a párhuzamos korpusz összes egyedi szava, a dokumentumok a forrásmondat és a hozzátartozó fordítás összefűzve. Az összefűzés azért valósítható meg, mert az algoritmus szózsákokkal dolgozik, ezért a szórend nem számít.

2.5.2. Annotátorok közötti egyetértés

Az alábbiakban bemutatok néhány módszert az annotátorok közötti egyetértés mérésére.

A Cohen-féle kappa két annotátor között mér egyetértést [44], amelyet az alábbi képlettel számolja ki:

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

, ahol a p_o a pontos egyezések mértéke, és p_e a véletlen folytán előálló egyetértés valószínűsége. A módszer két annotátor között tudja mérni a az egyetértést. Ha több annotátor van a Fleiss-féle kappa módszerét kell alkalmazni, amelynek képlete megegyezik a Cohen-féle kappa képletével, annyi különbséggel, hogy a p-t több annotátor kiértékeléséből számolja ki.

A Krippendorff alpha módszere a nem egyetértést veszi alapul [44] az alábbi képlettel:

$$\alpha = 1 - \frac{D_o}{D_e}$$

, ahol a D_o a megfigyelt nem egyetértések mértéke, míg D_e a nem egyetértés becsült valószínűsége.

DOI:10.15774/PPKE.ITK.2019.003

3. fejezet

A gépi fordítás kiértékelés elméleti háttere

3.1. Motiváció

A gépi fordítás használata széles körben elterjedt a mindennapokban, azonban a létező rendszerek között a fordítás minőségében jelentős különbségek mutatkoznak. Egyre több helyen merül fel igényként a gépi fordítás minőségének megállapítása. Cégek esetében igen nagy segítséget nyújthat egy minőségi mutató, ami nemcsak a gépi fordítás utómunkáját végző szakemberek tevékenységét támogathatja és gyorsíthatja, hanem segítheti a fordítócégeket költségeik csökkentésében is. Másik alkalmazási területét egy minőségi mérőszám létrehozása jelenti, a gépi fordítórendszerek kombinációjához. Megfelelő minőségbecsléssel több gépi fordítást össze tudunk hasonlítani, és a jobb fordítást kiválasztva, javíthatjuk rendszerünk végső minőségét. Végül, de nem utolsó sorban, ha ismerjük a fordítás minőségét, ki tudjuk szűrni a használhatatlan fordításokat, illetve figyelmeztetni tudjuk a végfelhasználót a megbízhatatlan szövegrészletekre.

A gépi fordítás minőségének automatikus mérése nem könnyű feladat. Alapvetően kétféle automatikus kiértékelési módszert különböztethetünk meg. Az első a referenciafordítással történő kiértékelés, amelyet hagyományos módszernek is mondhatunk. A második a referencia nélküli kiértékelő módszer, amelyet minőségbecslésnek hívunk.

DOI:10.15774/PPKE.ITK.2019.003

A hagyományos módszerek legnagyobb problémája, hogy referenciafordítást igényelnek, amelynek létrehozása igen drága és időigényes, ezért ezek a módszerek nem alkalmasak valós idejű használatra. A másik nagy probléma, hogy mivel ember által fordított referenciafordítás alapján készül a értékelés, a kiértékelés minősége jelentős mértékben függ a referenciafordítás minőségétől. Az elmúlt évek kutatásai [45] azt bizonyítják, hogy a hagyományos módszerek kiértékelései nem korrelálnak magasan az emberi kiértékelésekkel.

A minőségbecslő módszer nem igényel referenciafordítást, ezért valós időben is alkalmazható, és magasan korrelál az emberi kiértékeléssel. A minőségbecslés módszere a gépi tanulást használja a becsléshez alkalmazott modell felépítésére. A tanításhoz különböző jegyeket alkalmazunk, majd a jegyek segítségével a modellt emberi kiértékelésekre tanítjuk be. A rendszer erőssége, hogy a jegyek segítségével olyan problémákat is tudunk kezelni, amelyeket a hagyományos módszer nem képes kezelni.

Kutatásom során a minőségbecslés módszerét implementáltam angol-magyar nyelvre, majd alkalmaztam gépi fordítórendszerek kombinálására és egynyelvű szövegek minőségének megállapítására.

3.2. A gépi fordítás kiértékelésének szempontjai

A gépi fordítás kezdeti célja a tökéletes fordítás volt, de hamar rájöttek a kutatók, hogy ez igen nehéz feladat, ezért később célja a szöveg megértése (az idegennyelvű szöveg jelentésének átadása) és az információkinyerés (az idegennyelvű szöveg mondanivalójának megragadása) lett. Bár napjainkban már nem tűnik elérhetetlennek a tökéletes gépi fordítás, egyelőre az elsődleges cél mégsem az, hanem a szöveg megértése. Ezért a gépi fordítás kiértékelésében az elsődleges szempontok az alábbiak [46]:

- *Pontosság/tartalomhűség* (Adequacy): A pontossággal vagy tartalomhűséggel azt mérjük, hogy a gépi fordítás jelentésében mennyire felel meg a forrásnyelvi szöveg jelentésének. Ez nem azonos a referenciafordításnál használt pontosság (Precision) fogalmával (lásd 3.3. fejezet).

- *Gördülékenység/olvashatóság* (Fluency): A gördülékenységgel azt mérjük, hogy a célnyelvi fordítás önmagában, nyelvhelyesség szempontjából mennyire olvasható.

DOI:10.15774/PPKE.ITK.2019.003

- *Elfogadhatóság* (Acceptance): Az elfogadhatóság szubjektív mérték. Amikor egy személyt megkérnek arra, hogy értékeljen ki számomra egy fordítást, és ha az a fordítás minősége gyenge, akkor a kiértékelő személy nagy valószínűséggel nagyon rossz értéket ad majd rá. Azonban, ha megmondják, hogy azt a gyenge minőségű fordítást egy gép produkálta, akkor a kiértékelő személy nagy valószínűséggel kevésbé lesz szigorú, és jobb értékkel osztályozza ugyanazt.

Kutatásomban a kísérleteknél minden esetben jeleztem az emberi kiértékelők számára, hogy gépi fordítással dolgoznak, ezért a méréseim során a pontosság és a gördülékenység szempontokat vettem csak figyelembe.

3.3. Referenciafordítással történő kiértékelési módszerek

A referenciafordítással történő kiértékelési módszerek referenciafordítást használnak. A referenciafordítások emberek által fordított vagy javított szövegek. Vagyis adott egy forrásnyelvi szöveg, amit egy gépi fordítórendszer lefordít, majd emberek is lefordítják, vagy a gépi fordítást javítják ki. A hagyományos kiértékelő rendszer összehasonlítja a gép által lefordított szöveget az emberek által lefordított vagy javított szöveggel. Az összehasonlítás során mérni lehet a hasonlóságot vagy a különbözőséget a két szöveg között.

3.3.1. BLEU és BLEU-re épülő módszerek

A **BLEU** (BiLingual Evaluation Understudy) [47] az egyik legnépszerűbb hagyományos kiértékelő módszer. A BLEU metrika azt méri, hogy a gép által lefordított mondatokban lévő szavak és kifejezések mennyire pontosan illeszkednek a referenciafordításokhoz. A BLEU pontosságot mér, vagyis a számoláshoz az eredményhalmazt veszi alapul. A tartalomhűség és gördülékenység kezelésére a BLEU különböző n-grammokra számol pontosságot (P). Az unigrammal biztosítja a tartalom hűségét, míg az $n > 1$ n-grammokkal a gördülékenységet és a nagyobb pontosságot. Az algoritmus az n-grammokból számol súlyozott (w) mértani átlagot. A BLEU nem számol fedést, helyette bevezeti a „rövidség büntető” („brevity penalty” - BP) eljárást:

DOI:10.15774/PPKE.ITK.2019.003

$$BP = \begin{cases} 1 & \text{ha } c > r \\ e^{1-\frac{r}{c}} & \text{ha } c \leq r \end{cases}$$

, ahol c a lefordított mondat hossza, r a referencia mondat hossza, és N a leghosszabb n-gram n értéke. A képlet alapján minél rövidebb a lefordított mondat hossza a referenciamondat hosszához képest, annál nagyobb a „büntetés", vagyis kisebb a BLEU értéke. A végső BLEU metrikát az n-grammokból számolt átlag és a BP érték együtt adja eredményül:

$$BLEU = BP \ * \ exp(\sum_{n=1}^{N} w_n log P_n)$$

, ahol $exp = exponenciális függvény$.

A BLEU legnagyobb előnye, hogy több referenciát is tud kezelni, olcsó és gyors. Azonban nem érzékeny a szórendi átalakításokra, és ha két szó között csak toldalékban van eltérés, a BLEU két különböző szóként kezeli őket.

Az **OrthoBleu** algoritmus [48] karakteralapú n-grammokon számol F-mértéket. A módszer a ragozós nyelvek esetén, amikor két szó között csak a ragozásban van különbség, pontosabb eredményt ad a BLEU módszerhez képest. Az OrthoBleu egy karakterszintű javítási minőségi mutatószámot ad az utómunkát végző szakemberek számára, ami nagy segítség lehet számukra.

A **NIST** (NIST Metrics for Machine Translation - MetricsMATR) [49] a BLEU módszeren alapul, de pontosabb közelítést eredményez nála. Minden fordítási szegmenshez megadott módszerek alapján két független bírálatot rendel, majd ebből a két értékből állítja fel a végső pontszámot, amit hozzárendel minden fordítási szegmenshez. A NIST nem a referenciafordítást használja, hanem ezeket a bírálatok által kiszámolt pontszámokat. A NIST a szegmensekhez rendelt pontokból számol súlyozott átlagot, majd ezek kombinálásával számol egy dokumentumszintű pontszámot Ezután a dokumentumszintű pontszámokkal végez rendszerszintű kiértékelést. Az így kapott pontszámok és a bírálatok által számolt értékek közötti korreláció értéke adja a végső NIST mértéket. A NIST továbbá a ritkább n-grammokhoz nagyobb súlyt rendel, mivel a ritkább n-grammok informatívabbak, nagyobb információtartalommal bírnak.

DOI:10.15774/PPKE.ITK.2019.003

A **ROGUE** (Recall-Oriented Understudy for Gisting Evaluation) [50] a BLEU algoritmus ellentéte. Elsősorban automatikus szövegösszegző algoritmusok kiértékelésére használják, de gépi fordításhoz is alkalmazzák. A ROGUE algoritmus képlete teljes mértékben megegyezik a BLEU algoritmussal, annyi különbséggel, hogy a pontosság helyett fedést (C) számol, illetve a „brevity penalty" helyett „rövidség díjazó" („brevity bonus" - BB) van, vagyis a módszer azt „díjazza", ha az eredmény minél rövidebb:

$$ROGUE = BB * exp(\sum_{n=1}^{N} w_n log c_n)$$

3.3.2. METEOR, LEPOR, RIBES

Az alábbiakban bemutatok néhány fontosabb módszert, amelyek nem a BLEU módszerre épülnek, hanem a BLEU hiányosságaira próbálnak megoldást találni.

A **METEOR** (Metric for Evaluation of Translation with Explicit Ordering) [51] a BLEU módszer hibáit próbálja megoldani: a BLEU nem számol fedést, az $1 < n$ n-grammokat használja a gördülékenység számolására, nem kezeli a szavak pozícióját és mértani átlagot számol.

A METEOR metrika első része az unigrammokra számolt súlyozott F-mérték:

$$F_{átlag} = \frac{10PR}{R+9P}$$

, ahol P a pontosság, R a fedés. Látható, hogy a fedésre helyez nagyobb súlyt. A metrika második fele – hasonlóan a BLEU-höz – egy „büntetés" (penalty), de a METEOR úgynevezett csonkokat (chunks) használ a számoláshoz. A csonkok összefüggő szövegrészletek, amelyek megegyeznek a lefordított és a referencia mondatban. Például:

- fordítás: a gépi fordítás kiértékelése fontos feladat
- referencia: a gépi fordítás kiértékelése nagyon fontos feladat

A fenti példában 2 csonk (színessel jelölt szövegrészletek) van. Minél kevesebb a csonk, annál összefüggőbbek az egyezések, így annál kisebb a büntetés:

$$büntetés = 0,5 * \frac{csonkok\ száma}{unigram\ egyezések\ száma}$$

DOI:10.15774/PPKE.ITK.2019.003

A METEOR végső képlete:

$$METEOR = F_{átlag} \ * \ (1 - büntetés)$$

A METEOR továbbá WordNetet [41] használ a szinonimák kezelésére és tövesítő modult (porter stemmer).

A **LEPOR** (Length Penalty, Precision, n-gram Position Difference Penalty and Recall) [52] módszer a BLEU és a METEOR hiányosságaira keres megoldást. A METEOR nyelvspecifikus eszközöket használ, amitől a módszer nem lesz univerzális, valamint a módszer időigényes és komplex. A LEPOR igyekszik minden szempontot figyelembe venni: mondathossz, n-grammok, pontosság, fedés. Képlete:

$$LEPOR = LP \ * \ e^{NPD} \ * \ Harmonic(\alpha R, \beta P)$$

, ahol P a pontosság és R a fedés, továbbá:

$$LP = \begin{cases} e^{1-\frac{r}{c}} & \text{ha } c < r \\ 1 & \text{ha } c = r \\ e^{1-\frac{c}{r}} & \text{ha } c > r \end{cases}$$

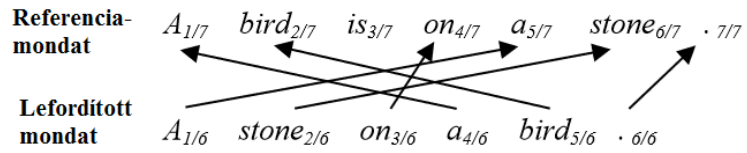
, ahol c a fordított mondat hossza és r a referenciamondat hossza. A fenti képlet alapján a módszer „büntetést” ad, ha a fordított mondat hosszabb vagy rövidebb a referenciamondatnál. Akkor nem „büntet”, ha egyforma hosszú a kettő. Továbbá az NPD az n-gram pozíciók „büntetése”:

$$NPD = \frac{1}{Ford_{hossz}} \sum_{i=1}^{Ford_{hossz}} |PD_i|$$

, ahol a $Ford_{hossz}$ a fordított mondat hossza és a PD_i a szóösszekötés pozíció különbsége (pozíciók „büntetése”). A pozíció „büntetésének” algoritmusa (lásd 3.1. ábra) a következő: először a lefordított mondatban és a referenciamondatban is index formában jelöljük a tokenek pozícióját a mondatban. Majd normalizáljuk az indexeket a mondathosszal. Végül vesszük az összekötött tokenek normalizált indexeinek különbségét.

DOI:10.15774/PPKE.ITK.2019.003

A képlet alapján, minél távolabb állnak az összekötött tokenek egymástól, annál nagyobb a „büntetés” mértéke.



$$NPD = \frac{1}{6} \times [\left|\frac{1}{6} - \frac{5}{7}\right| + \left|\frac{2}{6} - \frac{6}{7}\right| + \left|\frac{3}{6} - \frac{4}{7}\right| + \left|\frac{4}{6} - \frac{1}{7}\right| + \left|\frac{5}{6} - \frac{2}{7}\right|] = \frac{2}{7}$$

3.1. ábra LEPOR pozíciók büntetése [52]

A **RIBES** (Rank-based Intuitive Bilingual Evaluation Score) [53] módszert azokra a nyelvekre fejlesztették ki, ahol a szórend kiemelkedően fontos, és a forrás és a célnyelv között a szórend igen eltérő. A RIBES képlete:

$$RIBES = NKT * P^{\alpha} * BP^{\beta}$$

, ahol P az unigram pontosság, BP („brevity penalty”) megegyezik a BLEU rövidség büntető algoritmusával, $\alpha = \frac{1}{2}$, $\beta = \frac{1}{4}$ és $NKT(Normalizált\ Kendall\ \tau) = (\tau + 1) / 2$. Az NKT beindexeli a tokenek pozícióját a lefordított mondatban és a referenciamondatban, majd a két index sorozatára számol korrelációt.

3.3.3. WER, TER, HTER

Egy másik kiértékelési megközelítés, amikor a gépi fordítás és a referenciafordítás közötti különbözőséget vizsgáljuk. A különbözőséget vizsgáló algoritmusok esetén az eredmény értéke nagyobb, mint nulla, és minél kisebb az érték, annál jobb az eredmény.

A **WER** (Word Error Rate) egy szóalapú módszer, ami a különbözőséget méri a fordított mondat és a referenciamondat között. Az algoritmus a Levenshtein távolságon alapszik:

$$WER = \frac{S + D + I}{N}$$

, ahol S a cserék száma, D a törlések száma, I a beszúrások száma, $N = S + D + C$ (tokenek száma a referenciamondatban) és C a helyes szavak száma.

DOI:10.15774/PPKE.ITK.2019.003

A **TER** (Translation Edit Rate / Translation Error Rate) [54] az alapján számol fordítási hibaarányt a gépi fordítás és a referenciafordítás között, hogy mennyi javítást (token beszúrása, törlése, eltolása, helyettesítése) végeztek. A javítások számának és a referenciafordítás átlagos hosszának hányadosát képzi:

$$TER = \frac{javátások\ száma}{referenciafordítás\ szavainak\ átlagos\ száma}$$

A **HTER** (Human-targeted Translation Edit Rate / Human -targeted Translation Error Rate) a TER továbbfejlesztett változata. A TER nem kezeli a szemantikai problémákat. A gépi fordítás ugyanis csak azt számolja ki, hogy mennyi az eltérés a referenciafordítás és a gépi fordítás között, de lehet, hogy kevesebb javítással létrehozható olyan mondat, ami jelentésben megegyezik a referenciafordítással. Erre a problémára dolgozták ki a HTER módszert. A HTER módszer során célnyelvi anyanyelvű embereket kérnek fel, hogy minimális lépéssel javítsák ki a gépi fordító által generált mondatokat úgy, hogy megegyezzen a jelentése a referenciamondattal. Az így keletkezett új referenciamondatra számolják ki a TER értéket.

3.4. Minőségbecslés

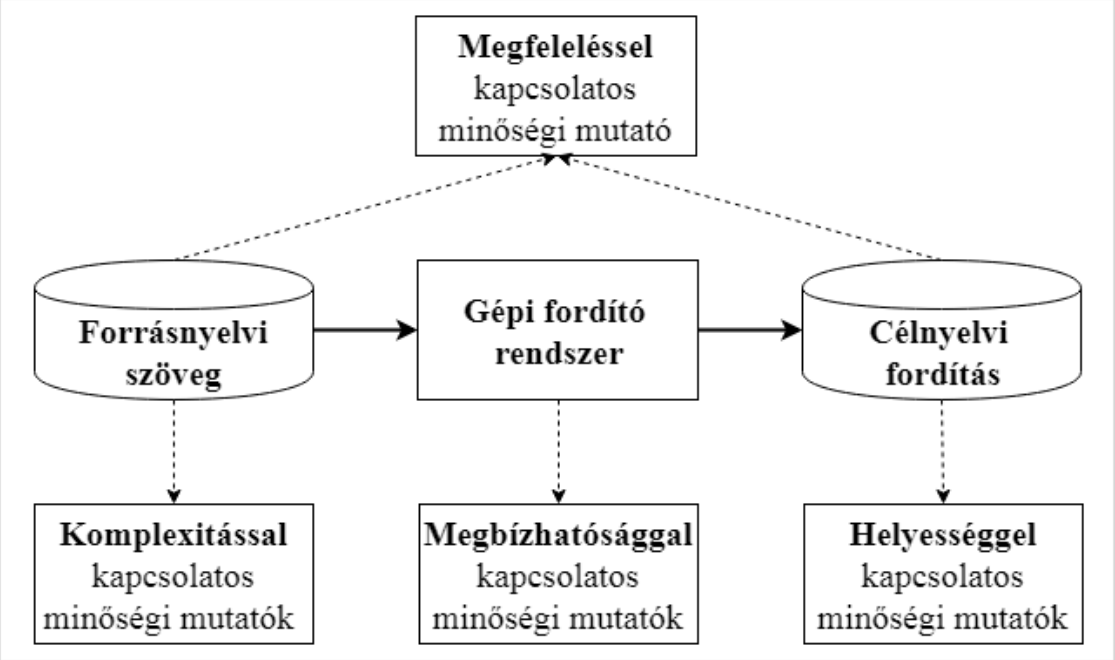
A minőségbecslés (Quality Estimation - QE) története 2005-ben kezdődött, amikor a Microsoft olyan kísérletet [55] kezdett el végezni, amiben azt vizsgálta, hogy különböző gépi fordítással kapcsolatos jegyek mennyire korrelálnak az emberi kiértékelésekkel. Különböző nyelvmodelleket és nyelvi elemzőket használtak a jegyek létrehozásában. 2007-ben Joshua és társai [56] a gépi tanulás módszerével kísérleteztek minőségbecslő modell megalkotására, de a jegyek létrehozásában még használtak referenciafordítással kapcsolatos módszereket. 2009-ben Lucia Specia [57] megalkotta a minőségbecslés (quality estimation) fogalmát. A minőségbecslés [58] egy gépi tanuláson alapuló referenciafordítás nélküli kiértékelő módszer, amely két részből áll: a *minőségbecslő modell tanítása* és a *minőség becslése*.

A minőségbecslés módszere a gépi tanuláson alapszik, ezért a minőségbecslő modell tanítása során egy gépi tanuló modellt kell betanítani. A tanítás két részből áll: a jegyek kinyerése és a modell felépítése.

DOI:10.15774/PPKE.ITK.2019.003

A *minőségbecslő modell tanításához* rendelkezésünkre állnak a forrásnyelvi és a gép által lefordított mondatok. Ezekből a szövegekből különböző nyelvfüggetlen és nyelvspecifikus minőségi mutatószámokat tudunk kinyerni, amelyek különböző jegyek (feature) segítségével történnek. A jegyeket kinyerhetjük a forrásnyelvi és a célnyelvi szövegből, valamint a gépi fordítórendszerből. Attól függően, hogy miből nyerjük ki a jegyeket, négy csoportba sorolhatjuk őket (lásd 3.2. ábra):

1. Komplexitással kapcsolatos jegyek: forrásmondatokból kinyert minőségi mutatók.
2. Helyességgel kapcsolatos jegyek: fordított mondatokból kinyert minőségi mutatók.
3. Megfeleléssel kapcsolatos jegyek: forrásnyelvi és fordított mondatok közötti viszonyából számított minőségi mutatók.
4. Megbízhatósággal kapcsolatos jegyek: gépi fordítórendszerből kinyert minőségi mutatók.



3.2. ábra Jegyek típusai

Egy másik szempont alapján, két kategóriába sorolhatjuk a jegyeket:

- *„Black-box"* jegyek: gépi fordítórendszertől független jegyek.

DOI:10.15774/PPKE.ITK.2019.003

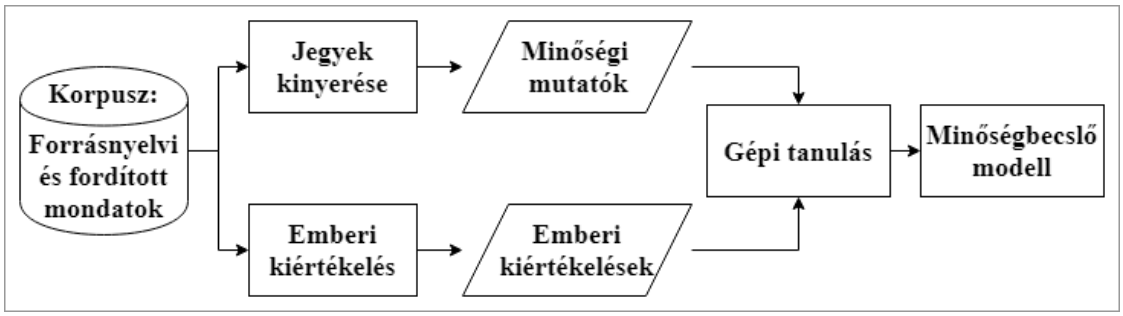
- *„Glass-box”* jegyek: gépi fordítórendszerből kinyert jegyek.

A jegyek segítségével kinyert minőségi mutatószámokkal tanítjuk be a minőségbecslő modellt.

A minőségbecslő modell tanítása egy gépi tanulás módszerével történik. A gépi tanuláshoz szükség van tanítóanyagra (megfigyelt mintákra) amire tanítjuk a modellt. A cél, hogy a becsült minőség magasan korreláljon az emberi kiértékeléssel, ezért a minőségbecslő modellt emberi kiértékelésekre tanítjuk be.

Az emberi kiértékelések a lefordított mondatok alapján készülnek. A kiértékelési szempontok lehetnek: tartalomhűség, gördülékenység, OK/BAD (egy fordítás elfogadható vagy eldobandó), HTER stb.

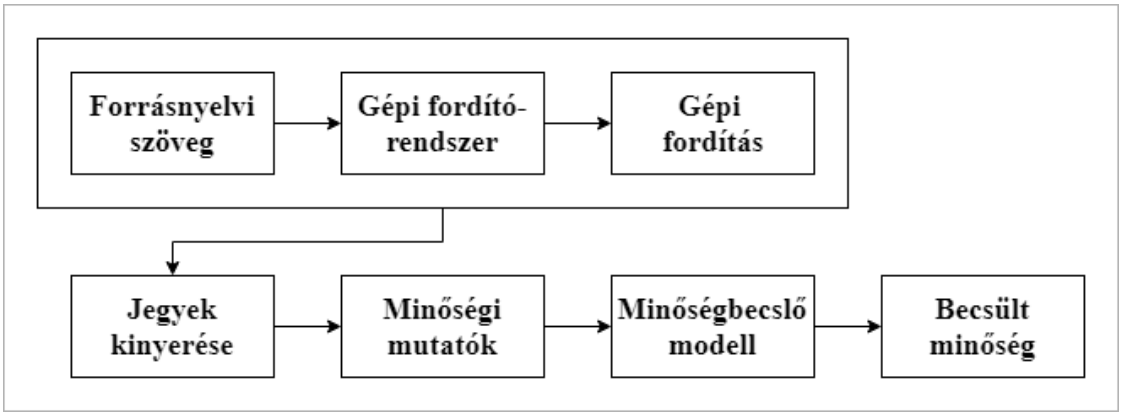
A végső minőségbecslő modell felépítéshez (lásd 3.3. ábra) felhasználjuk a jegyek által kinyert minőségi mutatókat, és betanítjuk az emberi kiértékelésekre.



3.3. ábra Minőségbecslő modell felépítése

A *minőség becslésének* (lásd 3.4. ábra) folyamata során először a jegyek segítségével kinyerjük a mutatószámokat, majd a minőségbecslő modell a jegyek által kinyert minőségi mutatók alapján végez minőségi becslést az új, ismeretlen bemeneti mondatokra.

DOI:10.15774/PPKE.ITK.2019.003



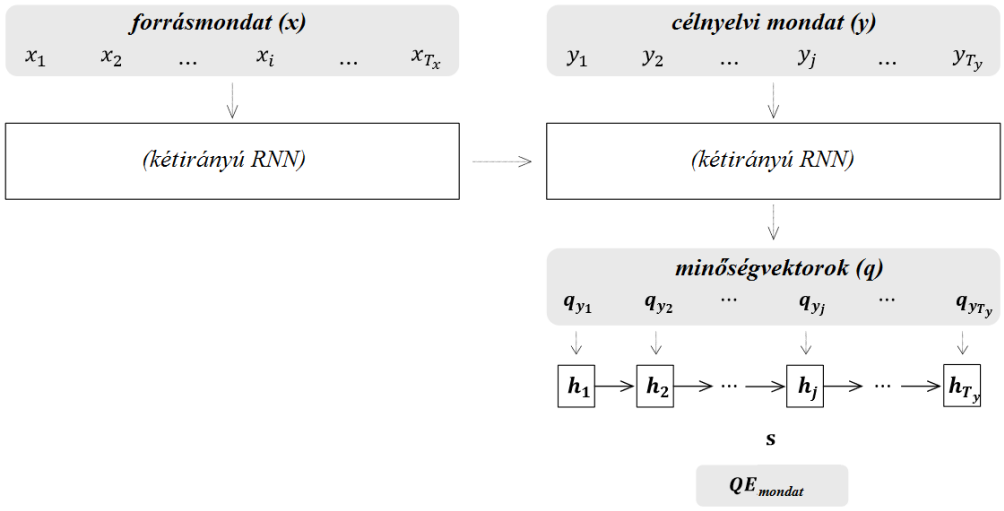
3.4. ábra Minőség becslésének folyamata

A minőségbecslés négy szinten történhet: szavak, kifejezések, mondatok és dokumentumok szintjén.

3.5. Neurális minőségbecslés

Az elmúlt két évben bontakozott ki a minőségbecslés neurális modellekkel való megközelítése. Az első áttörő eredményt 2016-ban publikálta Kim és Lee [59] POSTECH nevű rendszerükkel. Kutatásukban kétirányú RNN modelleket használtak a minőségbecsléshez (lásd 3.5. ábra). A kétirányú RNN abban különbözik az egyszerű RNN-től, hogy a feldolgozás kétirányú. Mindkét irányhoz tartozik egy-egy külön rejtett réteg. A két irány egymástól független. A kimeneti réteg mindkét rejtett réteget felhasználja. A módszer előnye, hogy ezzel egy időben korábbi, és későbbi kontextust is figyelembe vesz a modell. A POSTECH alapját az NMT-nél használt modellek képezik. Amíg az egyszerű NMT esetében a neurális modell a forrásnyelvi szegmensből becsüli a célnyelvi fordítást, addig a POSTECH módosítva, a forrásnyelvi mondatból és a célnyelvi fordításból együtt, kétirányú RNN modellek segítségével, minőségvektorokat hoz létre. Ezek a minőségvektorok egy újabb RNN modell bemeneti adatai lesznek, amellyel megbecsülik a fordítás minőségét.

DOI:10.15774/PPKE.ITK.2019.003

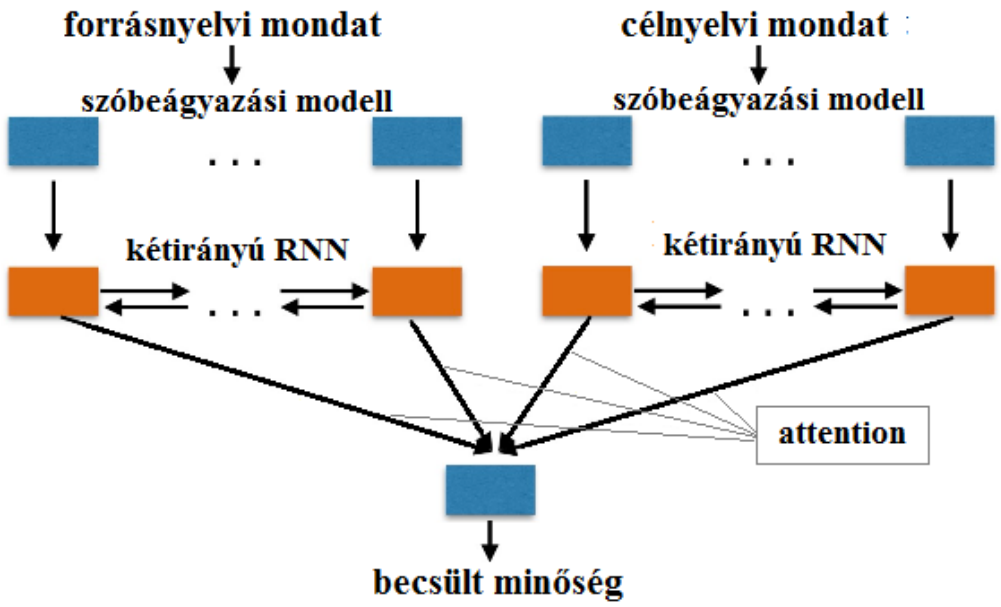


3.5. ábra POSTECH minőségbecslő modell [59]

A modellek tanításához nagy mennyiségű párhuzamos adat szükséges. Első lépésben be kell tanítani a forrásmondathoz és a célnyelvi mondathoz tartozó kétirányú RNN modelleket. Ez a folyamat az NMT modellek tanításához hasonlóan nagyméretű párhuzamos korpusszal történik. Második lépésként az első lépésben betanított modell és egy ember által kiértékelt korpusz segítségével betanítják a minőségbecslő RNN modellt.

A POSTECH rendszer nem volt nyilvános, ezért Ive és társai újraimplementálták, és elérhetővé tették deepQuest keretrendszer néven [60]. Emellett készítettek egy saját neurális minőségbecslő modellt, a BI-RNN modellt [60]. Kutatásukban két darab kétirányú RNN modellt használnak (lásd 3.6. ábra). A tanítás során, az emberi kiértékeléssel rendelkező párhuzamos korpusz segítségével, az NMT-hez hasonlóan, betanítanak egy enkódert. Az enkóderben a forrásnyelvi és a célnyelvi RNN modellt egymástól függetlenül tanítják be. A két RNN modell kimenetét konkatenálva hoznak létre egy reprezentációt, aminek segítségével megbecsülik a fordítás minőségét. A minőségbecsléshez nem tanítanak be egy külön modellt, hanem az „attention” mechanizmus segítségével teszik ezt meg. Az „attention” módszer [61] során a modell a fordítás közben nem csak az enkóder legutolsó kimenetét veszi számításba, hanem az addigi gyűjtött összes keletkező információt összefoglalja, és ezt az információt is figyelembe veszi. A döntéshozatalban e plusz vektor mint információ nyújt segítséget.

DOI:10.15774/PPKE.ITK.2019.003



3.6. ábra BI-RNN minőségbecslő modell [60]

Jelenleg a legjobb minőséget elérő módszer a QE Brain [62]. A QE Brain két részből áll: jegykinyerő modul és minőségbecslő modul. A jegykinyerő moduljukat „Bilingual Expert" modellnek hívják, amely három modulból áll: egy „transformer self-attention" [63] (a „self-attention" abban különbözik az „attention" modelltől, hogy a plusz információt önmagából és környezetéből nyeri ki) alapú enkóder a forrásnyelvi szöveg számára, egy „transformer self-attention" alapú dekóder a célnyelvi szöveg számára, és egy szóvisszaállító modul szintén a célnyelvi szöveg számára. A modell a három modulból állít elő egy jegyvektort, majd ennek a jegyvektornak a segítségével becsüli meg a fordítás minőségét. A minőségbecslő modult egy kétirányú „Hosszú Rövid Távú Memória" („Long Short Term Memory" - LSTM) modell [64] valósítja meg. Az LSTM modell segít abban, hogy csökkentse az RNN modell visszacsatolása során okozta gradiens robbanását.

DOI:10.15774/PPKE.ITK.2019.003

4. fejezet

A HuQ korpusz

4.1. Előzmények

A gépi fordítás elterjedésével fontos feladat lett a gépi fordítás minőségének megállapítása. A gépi fordítás minőségének becslése fontos kutatási terület lett a nyelvtechnológiában, fontosságát abból is láthatjuk, hogy 2012 [65] óta minden évben kiírnak versenyeket (megosztott feladatokat - Shared Task) a gépi fordítás minőségbecslés témakörben, hogy megtalálják a legjobb módszert. A kiosztott feladatokban tanító és teszt korpuszokat biztosítanak a kutatók számára. A korpuszokban lefordított mondatok állnak, amelyeket emberek értékeltek ki. Az emberi kiértékelések lehetnek: HTER, METEOR és utószerkesztésre ráfordított munka mértéke. De sajnos a mai napig [66], az általam készített korpuszon kívül, nem áll rendelkezésünkre ember által kiértékelt angol-magyar nyelvű korpusz.

A jelen kutatásomban készítettem egy kézzel kiértékelt korpuszt, angol-magyar minőségbecslő rendszer betanításához és teszteléséhez. Az elkészült korpuszt HuQ (Hungarian Quality estimation) korpusznak neveztem el.

4.2. Kapcsolódó munkák

Az *Association for Computational Linguistics (ACL)* 2012 óta rendez konferenciát és versenyt a minőségbecslés témájában, hogy megtalálja a legjobb minőségbecslő módszert [65]. A versenyhez biztosítottak egy angol-spanyol tanítókorpuszt, amely 1832

DOI:10.15774/PPKE.ITK.2019.003

mondatpárból áll, valamint egy tesztkorpuszt, amely 422 mondatpárból állt. Minden fordítást 3 résztvevő értékelt ki, 1-től 5-ig terjedő Likert skálán. A kiértékelés szempontja az volt, hogy mennyire hibás a mondat, mennyire szükséges javítani a gépi fordítást. 2013-ban [72] már több feladatot írtak ki, amelyből az egyik feladat kiértékelési szempontja a HTER volt. Emellett egy másik feladatban minden forrásmondatot 5 különböző gépi fordító rendszerrel lefordítottak, majd a minőségük alapján a résztvevőkkel sorba rendeztették az egyes forrásmondathoz tartozó fordításokat. Ebben a feladatban olyan algoritmust kellett írni, amely a legmagasabb rang korrelációt eredményezte. 2014-ben [73] annyi változott az előző évekhez képest, hogy kiírtak egy olyan feladatot, amelyben a korpuszban az egyes forrásmondatokhoz 3 különböző fordítás tartozott: egy emberi fordítás, egy statisztikai és egy szabály-alapú gépi fordítás. Minden fordítást egy 1-3 terjedő osztályattribútum jellmezte, amelyekkel azt jelölték, hogy mennyi munka szükséges a fordítás javításához.

Kutatásomban a saját tanítóanyagomat a versenyeken biztosított korpuszok elkészítésének módjait figyelembe véve hoztam létre . Különböző típusú gépi fordító rendszereket használtam fel a fordításaim előállításához. Emberi kiértékelésnél a Philipp Koehn által ajánlott 1-5 Likert skálát, valamint annak két szempontját (tartalomhűség és gördülékenység) vettem figyelemben [16]. Az osztályattribútumok előállításánál a 2014-es versenyen biztosított korpusz annotálását vettem alapul.

4.3. A HuQ korpusz bemutatása

A HuQ korpusz elkészítéséhez vettem 300 angol-magyar mondatpárt a Hunglish korpuszból [67], melynek angol mondatait lefordítottam 4 különböző gépi fordítórendszerrel:

- MetaMorpho szabályalapú gépi fordítórendszer [68],
- Google Fordító[1],
- Bing Fordító[2],
- MOSES statisztikai gépi fordító keretrendszer [69].

[1] https://translate.google.hu/
[2] https://www.bing.com/translator

DOI:10.15774/PPKE.ITK.2019.003

A négy gépi fordítással és a korpuszban lévő emberi fordítással együtt így összesen 1500 mondatpárból áll a HuQ korpusz.

Amikor 2016-ban készítettem [8] a korpuszt, még a Google Fordító és a Bing Fordító is statisztikai fordítórendszerek voltak. De 2016 óta a Google Fordító neurális gépi fordítórendszer lett [70], és a Microsoft is kísérletezik a neurális hálózaton alapuló gépi fordítási modellekkel[3].

A MetaMorpho egy szabályalapú gépi fordítórendszer, amely a HUMOR (High-speed Unification MORphology) morfológiai elemzőre [71] épül. A MOSES egy nyílt forráskódú gépi fordító keretrendszer.

Kutatásomhoz a MOSES keretrendszerrel betanítottam egy angol-magyar gépi fordítórendszert. A tanításhoz a Hunglish korpuszt [67] használtam, amely ∼1,1 millió angol-magyar fordított mondatpárt tartalmaz. Természetesen kivettem azokat a mondatokat, amelyeket felhasználtam a HuQ korpusz elkészítéséhez.

A Hunglish korpusz vegyesen tartalmaz feliratokat, jogi szövegeket, szoftverdokumentációkat és irodalmi szövegeket. A feliratok hétköznapi mondatok, amelyek sok szlenget tartalmaznak. Az irodalmi szövegek komplex nyelvtani szerkezetűek, sok idegen szóval. A jogi szövegek szintén bonyolult nyelvtani szerkezetűek, amelyek távol állnak a hétköznapi mondatoktól. A szoftverdokumentációk sok rövid mondatot tartalmaznak.

Mivel a minőségbecslő modellhez emberi kiértékelésekre van szükség, ezért mind az 1500 fordítást 3 annotátorral értékeltettem ki. A felkért három annotátor közül az egyik nyelvész (L), a másik gépi fordítás szakértő (M), a harmadik pedig nyelvtechnológus (T).

A fordítások kiértékeléséhez használtam közösségi közreműködést (crowdsourcing) is, de a jelen kutatásban nem használom őket. Így követni és ellenőrizni tudtam az annotátorok munkáját. Ha felmerült valami félreértés, azt meg tudtuk beszélni. Az annotátorok a kutatás megkezdése előtt kiértékeltek 50 próbafordítást, amelyeket kézzel válogattam, hogy alkalmasak legyenek a tipikus problémák megbeszélésére. Ez az 50 mondat nem szerepel a kutatásban használt HuQ korpuszban.

A kiértékelés 2 szempontból állt (lásd 4.1. táblázat): *tartalomhűség* és *gördülékenység*.

[3] https://translator.microsoft.com/neural/

DOI:10.15774/PPKE.ITK.2019.003

Tartalomhűség	Gördülékenység
1: egyáltalán nem jó	1: érthetetlen a mondat
2: jelentésben egy kicsit pontos	2: nem helyes a mondat
3: közepesen jó a pontosság	3: több hibát tartalmaz a mondat
4: jelentésben nagyrészt pontos	4: majdnem jó a mondat
5: jelentésben tökéletesen pontos	5: hibátlan a mondat

4.1. táblázat Értékelési szempontok

Gépi fordító-rendszer	**Példamondat**	**Tartalom-hűség**			**Gördülé-kenység**		
		L	**M**	**T**	**L**	**M**	**T**
Forrás	Smith turned the question over in his mind.						
Referencia	Smith megvizsgálta a kérdést.						
MetaMorpho	Smith a kérdést forgatta a fejében.	2	5	5	4	5	5
Google (SMT)	Smith megfordult a kérdés felett a fejében.	1	3	5	5	3	4
Bing (SMT)	Smith megfordult a kérdés a fejében.	4	5	4	4	4	4
MOSES	Cyrus smith a kérdést.	1	1	1	1	1	4

4.2. táblázat Példa a különböző gépi fordításokra

A kiértékelés skálájához a Koehn által ajánlott [16] Likert skálát vettem alapul. Az annotátorok a két megadott szempont alapján 1-től 5-ig értékelhették ki a fordításokat. A tartalomhűség értékelésénél az annotátorok figyelembe vették mind a forrás, mind a gépi fordítást, míg a gördülékenység esetében csak a gépi fordítást értékelték. Mind a három annotátor magyar anyanyelvű, és legalább B2 szintű angol nyelvvizsgával rendelkezik.

A 4.2. táblázatban látható példa a különböző gépi fordítások különbségeire mutat rá. A Google (SMT) és a Bing (SMT) statisztikai gépi fordítórendszerek, amelyeknek az a nagy előnyük, hogy óriási korpuszon tanították őket. Ugyanakkor, ha egy kifejezés nem szerepel a korpuszban, akkor csökken a valószínűsége, hogy pontos fordítást ad. Ebben a példában láthatjuk, hogy jelentésben nem azt eredményezte a két rendszer, amit vártunk volna. Ha viszont nem vesszük figyelembe azt, hogy nyelvtanilag mennyire helyes a mondat, akkor nem áll messze egyik fordítás sem a referenciafordítástól.

A MOSES rendszer, ezzel ellentétben, nagyon rossz fordítást készített. Ez alátámasztja azt a tényt, hogy a tanítókorpusz mérete nem elég nagy egy ilyen nehezebb mondat fordítására.

DOI:10.15774/PPKE.ITK.2019.003

A MetaMorpho szabályalapú rendszer példájából az látszik, hogy nyelvtanilag helyes a mondat, ugyanis a rendszer morfológiai és szintaktikai elemzést is végez a fordítás előállításához. Mivel ez a rendszer sem használ szemantikai elemzést, ezért csak a szavak alapjelentéseit használja a fordításhoz, és nem tudja teljes mértékben azt a fordítást eredményezni, amit várnánk.

A 4.2. táblázatban látható példában a probléma forrása az, hogy jelentésében nem Smith „fordult”, hanem a kérdés „fordult”, továbbá ennek a kifejezésnek igazából nem sok köze van a „fordul” szóhoz, ehelyett itt ez egy idióma. Viszont a jelentés egy olyan problémakör, amely a többértelműségével az embereket is megosztja. A példában az is látható, hogy az emberi kiértékelések némely esetben igen eltérőek. Jellemzően a nyelvész annotátor sokkal szigorúbb volt a kiértékelés során, míg a gépi fordító és a nyelvtechnológus szakértők sokkal engedékenyebbek voltak. Azért, hogy kiegyenlítsem a nagy különbözőséget, az egyes mondatok kiértékeléséhez a három annotátor értékének a számtani átlagát vettem. Külön kiszámoltam a tartalomhűségre adott értékek átlagát, külön a gördülékenységre adott értékek átlagát, majd a fordítások végső minőségéhez a kiszámolt átlagoknak az átlagát vettem. Az alább felsoroltak a három említett átlag:

- TA: tartalomhűség értékek számtani átlaga;
- GA: gördülékenység értékek számtani átlaga;
- TG: TA és GA értékeinek átlaga.

4.3.1. Osztályozási modell

Cégek esetében sokszor csak arra az információra van szükség, hogy az adott gépi fordítás jó vagy rossz; használható vagy eldobandó. Ezért én is létrehoztam a folytonos értékekből osztályokat. Kétféle korpuszt készítettem. Az egyik három osztályattribútumból áll, a másik kettő osztályattribútumból.

A három osztályattribútumból álló korpusz osztályai:

- BAD (rossz, eldobandó fordítás): $1 \leq x \leq 2$,
- MEDIUM (közepes, javítandó fordítás): $2 < x < 4$,
- GOOD (jó, használható fordítás): $4 \leq x \leq 5$.

DOI:10.15774/PPKE.ITK.2019.003

A kettő osztályattribútumból álló korpusz osztályai:

- ER (hibás, javítandó fordítás): $x \leq 4$,
- OK (jó, nem javítandó fordítás): $x > 4$.

Az így létrehozott osztályokból az alábbi osztályozási mértékeket alkottam:

- CLTA: TA-ból létrehozott osztályattribútumok (három osztályattribútumos),
- CLGA: GA-ból létrehozott osztályattribútumok (három osztályattribútumos),
- CLTG: TG-ből létrehozott osztályattribútumok (három osztályattribútumos).
- CLBITA: TA-ból létrehozott osztályattribútumok (bináris osztályozó),
- CLBIGA: GA-ból létrehozott osztályattribútumok (bináris osztályozó),
- CLBITG: TG-ből létrehozott osztályattribútumok (bináris osztályozó).

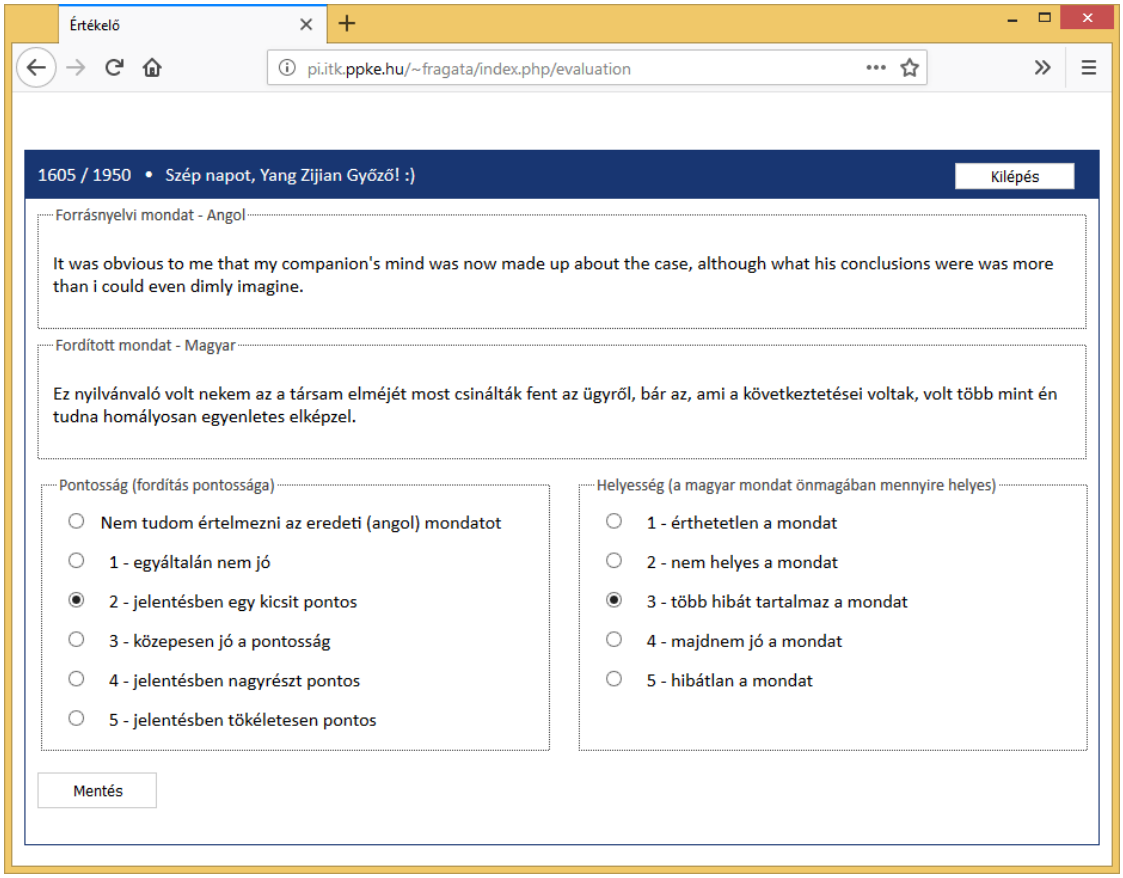
Így a HuQ korpusz szegmenseire a folytonos értékek mellett osztályattribútumokat is meghatároztam. Az angol-magyar minőségbecslő rendszer létrehozásakor külön modelleket tanítottam be az osztályattribútumokra.

4.4. Mérések és eredmények

Az annotátorok munkájának segítésére létrehoztam egy weboldalt[4] (lásd 4.1. ábra), amin ki tudták értékelni a mondatokat. A weboldalon található a forrásmondat és a hozzátartozó fordítás. Az annotátorok két szempont alapján tudták leadni a kiértékelésüket, 1-től 5-ig: Pontosság = Tartalomhűség; Helyesség = Gördülékenység. A felületen található egy „Nem tudom értelmezni az eredeti (angol) mondatot” választható lehetőség is. Ez arra az esetre van, ha valaki nem tudta értelmezni a forrásmondatot, ilyen esetben a program nem veszi figyelembe ezt a kiértékelést. Ezzel kerülöm el a hibás kiértékeléseket, vagyis csökkentem a zajok lehetőségét. A kutatásomban a három annotátor egyszer sem használta ezt a funkciót.

[4] http://nlpg.itk.ppke.hu/projects/huq

DOI:10.15774/PPKE.ITK.2019.003



4.1. ábra A kiértékelő weboldal

Először a HuQ korpusz tulajdonságait vizsgáltam meg. Megmértem az annotátorok közötti egyetértést (lásd 4.3. táblázat). Ahogy vártam, az annotátorok közötti egyetértés nem volt magas. Ezért is fontos, hogy az átlagát vettem a három annotátornak. Azonban a 4.4. táblázatban látható, hogy a kiértékelések korrelációi magasak. Vagyis annak ellenére, hogy a nyelvész szigorúbb volt, a másik két annotátorhoz hasonló véleményen volt.

	TA	GA	TG	CLTA	CLGA	CLTG
Fleiss-féle Kappa	0,357	0,463	0,315	0,44	0,521	0,493
Krippendorff Alpha	0,357	0,463	0,316	0,44	0,521	0,493
Páronkénti Cohen Kappa	0,360	0,464	0,317	0,444	0,522	0,494
Páronkénti átlag	52,5%	61,2%	43,8%	70,1%	74,4%	70,6%

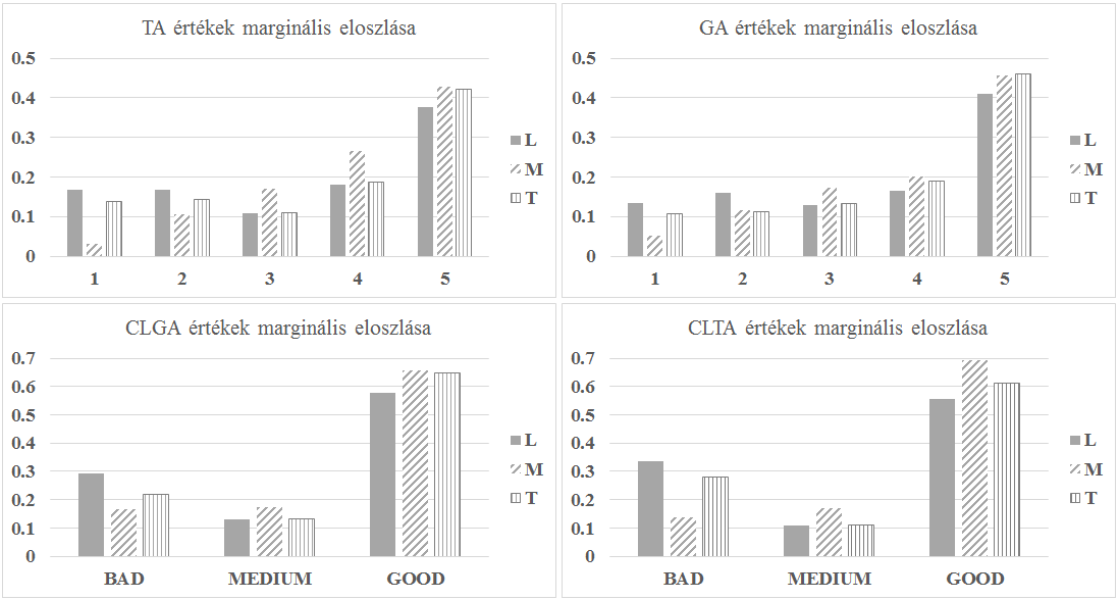
4.3. táblázat Annotátorok közötti egyetértés

DOI:10.15774/PPKE.ITK.2019.003

	TA	GA	TG
L és M értékeinek korrelációja	0,6994	0,7784	0,7857
M és T értékeinek korrelációja	0,6823	0,7446	0,7539
T és L értékeinek korrelációja	0,7819	0,8408	0,8469

4.4. táblázat Annotátorok kiértékeléseinek korrelációi

A HuQ korpusz kiértékelés mértékeinek eloszlását (lásd 4.2. ábra) tekintve, az látható, hogy jó minőségű fordításokból van a legtöbb. Ez azért van, mert a korpusz tartalmaz emberi fordításokat, amelyek jó minőségűek, és emellett a gépi fordítórendszerek is produkáltak jó minőségű fordításokat. Továbbá a gépi fordító szakértő ritkábban adott 1-es értéket. Ez az elfogadhatóság mértékének következménye, hiszen egy gépi fordítással foglalkozó szakértő számára a cél a megértés és információkinyerés, és nem a tökéletes fordítás volt, ezért ritkábban adta az 1-es értéket. Ezzel szemben a nyelvész sokkal inkább adott szélsőségesebb értékeket, mint közepest.



4.2. ábra A kiértékelések marginális eloszlása

Megvizsgáltam és összehasonlítottam – az emberi kiértékelések alapján – a gépi fordítórendszerek minőségét (lásd 4.5. táblázat). Ahogy sejteni lehetett, a MOSES gépi fordítórendszer teljesített a leggyengébben, a MetaMorpho szabályalapú gépi fordítórendszer pedig a legjobban. Ez azért lehet, mert a jogi és irodalmi komplexebb mondatok eseté-

DOI:10.15774/PPKE.ITK.2019.003

ben a nyelvi (morfológiai és szintaktikai) elemzővel rendelkező, szabály alapú rendszer pontosabb fordítást eredményez, mint a csupán statisztikai alapon működő Google és Bing fordítók.

	TA átlag	GA átlag	TG átlag
MetaMorpho	3,8707	3,8651	3,8679
Google (SMT)	3,6395	3,5729	3,6062
Bing (SMT)	3,2166	3,2256	3,2211
MOSES	3,0175	3,1872	3,1024

4.5. táblázat Gépi fordítórendszerek összehasonlítása

A HuQ korpusz TA, GA, TG, CLTA, CLGA és CLTG mértékei segítségével különböző minőségbecslő modelleket hoztam létre. A minőségbecslő modell felépítéséhez több, mint száz jeggyel kísérleteztem (részletes kifejtés az 5. fejezetben).

Megvizsgáltam, hogy a korpusz méretének növekedésével milyen mértékben javul a minőségbecslő modell minősége. A HuQ korpuszból az alábbi részkorpuszokat készítettem:

- TG-100: 100 mondatpárból álló részkorpusz,
- TG-500: 500 mondatpárból álló részkorpusz,
- TG-1000: 1000 mondatpárból álló részkorpusz,
- TG-1500: teljes HuQ korpusz.

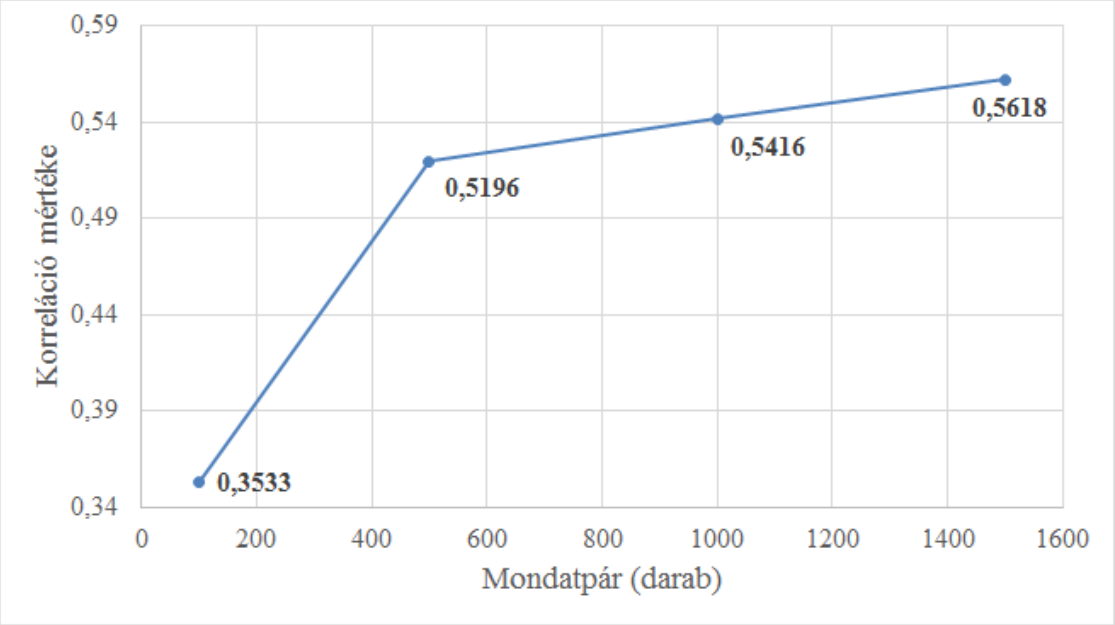
Validálás végett, a TG-100, a TG-500 és a TG-1000 esetében, a részkorpusz mondatpárjai véletlenszerűen lettek kiválasztva. tízszer végeztem el a mérést, a 4.6. táblázatban a 10 mérés átlaga látható.

	Korreláció ↑	MAE ↓	RMSE ↓
TG-100	0,3533	0,9229	1,1505
TG-500	0,5196	0,8621	1,0692
TG-1000	0,5416	0,8044	1,0315
TG-1500	**0,5618**	**0,7962**	**1,0252**

4.6. táblázat Korpusz méretének növelése

DOI:10.15774/PPKE.ITK.2019.003

A minőségbecslő modell kiértékeléséhez a Pearson-féle korrelációt, a MAE-t és az RMSE mértékeket használtam. A mérés eredményeiből (lásd 4.6. táblázat) az látható, hogy a méret növekedésével javul a kiértékelés minősége is. Azonban a korpusz méretének növekedése és a minőség növekedése nem egyenesen arányosak (lásd 4.3. ábra). Vagyis az eredményből az feltételezhető, hogy a HuQ korpusz további növekedésével nem okozna jelentős minőségbeli javulást. Ezért a kutatás szempontjából az 1500 mondatpár elégséges a további kísérletek és kutatások elvégzésére.



4.3. ábra A korpusz méretének növekedése és a korreláció változásának függvénye

4.5. Továbblépési lehetőségek

A jelen kutatásban tárgyalt HuQ korpusz 1500 mondatpárral rendelkezik, melyeket három annotátor értékelt ki. A feladathoz közösségi közreműködést (crowdsourcing) is kértem. Jelenleg 1950 mondat van kiértékelve (benne van a HuQ korpusz is). Minden mondatot maximum három ember értékelheti ki, körülbelül 450 olyan fordítás van jelenleg, amelyet – a három annotátoron kívül – ismeretlen emberek értékelték ki. Azonban a 450 fordításból nem mindegyiket értékelte ki három ember. Ezért sajnos még nem tudok

DOI:10.15774/PPKE.ITK.2019.003

további kutatásokat végezni. Ha ez a szám tovább nő, újabb méréseket tudok végezni rajta. Érdekes lehet megvizsgálni azt is, hogy a három annotátoron kívüli kiértékelések milyen egyetértési értékeket mutatnak, vagy az értékelések milyen eloszlást mutatnak.

4.6. Összegzés

A minőségbecslés módszerét még nem kutatták angol-magyar nyelvre, ezért nem készült angol-magyar nyelvű tanítókorpusz sem.

A kutatásom során minőségbecslő modell tanítására létrehoztam egy angol-magyar korpuszt. A korpusz létrehozásához vettem 300 angol nyelvű mondatot, amelyeket lefordítottam 4 különböző gépi fordítórendszerrel. A korpusz továbbá tartalmaz emberi fordítást is, így összesen 1500 mondatpárból áll. Mind az 1500 mondatpárt három annotátor értékelte ki, Likert skálán, két szempont alapján, 1-től 5-ig: tartalomhűség és gördülékenység. Az annotátorok kiértékelésének segítségére létrehoztam egy weboldalt.

Az így létrehozott HuQ korpuszra végeztem statisztikai és annotátorok közötti egyetértési méréseket. A három annotátor különböző szempontokból értékelte ki a mondatokat, ezért a három kiértékelés számtani átlagát vettem az egyes szegmensekre. Továbbá, az osztályozási feladathoz készítettem osztályattribútumokat a kiértékelésekből.

Ezután megvizsgáltam, hogy a korpusz növekedése milyen hatással van a rendszer minőségére. Azt a következtetést vontam le, hogy mivel a korpusz méretének növekedése a minőség növekedésével nem egyenesen arányos (logaritmus függvényt ír le), így a kutatásom szempontjából az 1500 mondatpár elégséges.

Kapcsolódó tézis

1. tézis: **Létrehoztam egy kézzel kiértékelt korpuszt, amely angol-magyar nyelvű minőségbecslő rendszer tanítására alkalmas.**

A tézishez kapcsolódó publikációk: [6] [8].

DOI:10.15774/PPKE.ITK.2019.003

5. fejezet

A Hun-QuEst rendszer

5.1. Előzmények

A minőségbecslés módszerét angol-magyar nyelvre még nem kutatták előttem. Jelenleg négy szinten zajlik a kutatás a minőségbecslés területén [66]: szavak, kifejezések, mondatok és dokumentumok szintjén.

Az első úttörő kutatást a minőségbecslés területén Specia és társainak kutatásai [57] jelentették. Kutatásukban a gépi tanulás módszerét alkalmazták a gépi fordítás minőségének becslésére. Módszerük: forrásmondatokból és a gépi fordításokból különböző jegyek segítségével minőségi mutatókat nyertek ki, majd a minőségi mutatókkal betanítottak egy gépi tanuló algoritmust.

A jegyek kinyeréséhez létrehoztak egy szabadon hozzáférhető JAVA nyelven implementált keretrendszert, a QuEst-et [58]. A rendszer továbbfejlesztett verziója a QuEst++ [74]. A jegyek kinyerése után, egy gépi tanuló algoritmussal lehet a kinyert jegyeket emberi kiértékelésekre betanítani. A gépi tanuláshoz Specia és társai Python nyelven írtak egy szoftveres megoldást.

Kutatásomban a jegyek kinyeréséhez a QuEst keretrendszert használtam. A könnyebb integrálhatóság végett, a gépi tanuláshoz a JAVA nyelven implementált Weka [75] gépi tanuló szoftvert használtam.

A QuEst keretrendszer elkészítése mellett Beck és társai, angol-spanyol nyelvre, több, mint 160 jeggyel kísérleteztek [76]. Kutatásukban az emberi kiértékelések HTER értékek voltak. Bebizonyították, hogy – a „kevesebb néha több” elvén – nem minden jegy releváns

DOI:10.15774/PPKE.ITK.2019.003

az eredményre nézve, és kevesebb releváns jeggyel jobb eredményt lehet elérni, mint a 160 jeggyel betanított modellel. Létrehoztak egy alapjegykészletet (baseline features)[1], amely 17 nyelvfüggetlen „black-box" jegyet tartalmaz.

A gépi tanuló algoritmusok kiválasztásának kutatása terén 2016 előtt a szupport vektor gépek, a szupport vektor regresszió és a Gauss-eljárás módszerek domináltak és érték el a legjobb eredményeket [77, 78]. 2016-tól a neurális hálózaton alapuló módszerek már egyértelműen jobb eredményeket produkáltak [78, 79].

Kutatásomban először létrehoztam egy angol-magyar minőségbecslő rendszert, majd új szemantikai jegyekkel kísérleteztem. Kipróbáltam több különböző gépi tanuló algoritmust, és végeztem optimalizációt is, amellyel kevesebb jeggyel tudtam jobb eredményt elérni. Amikor a kutatásomat végeztem nem a neurális hálózat alapú minőségbecslés módszere dominált a kutatásokban, ezért a neurális hálózatot jegyek létrehozásához használtam. A neurális hálózat alapú minőségbecslő modell kutatása angol-magyar nyelvre kezdeti fázisban van.

5.2. Kapcsolódó munkák

2012 óta rendeznek konferenciát és versenyt a minőségbecslés témájában, hogy megtalálják a legjobb minőségbecslő módszert [65]. Kezdetben a versenyen csak mondatszintű feladat volt, de ma már négy kategóriában lehet nevezni. Első a mondatszintű minőségbecslés, második a szószintű, harmadik a kifejezésszintű és negyedik a dokumentumszintű. Mondatszinten a HTER metrikát használták az emberi kiértékeléshez, szó és kifejezés szinten azt mérték, hogy hány szó vagy kifejezést fordított jól vagy rosszul a gép. A dokumentumszintű kiértékelés egy új irány, amiben a korpusz szószintű kiértékelést tartalmaz, és ennek segítségével kell dokumentumszintű minőségbecslést végezni.

Az elmúlt évek során a minőségbecslés témájában három fő irányban folynak kutatások. Az egyik irány az új releváns jegyek felfedezése [76]. A másik irány a jegykészlet optimalizálása gépi tanulás módszerek kísérletezésével [88, 89]. A harmadik irány a neurális hálózat alapú minőségbecslés iránya [59, 60]. Utóbbiban jelenleg a QE Brain [62] a legeredményesebb rendszer.

[1] https://www.quest.dcs.shef.ac.uk/quest_files/features_blackbox_baseline_17

DOI:10.15774/PPKE.ITK.2019.003

Munkám során mindhárom irányban végeztem kutatásokat. A gépi tanulás módszereinek kísérletében megvizsgáltam, hogy melyik módszer alkalmas leginkább az angol-magyar minőségbecslés számára. Fő kutatási irányom az új releváns jegyek kutatása volt. Létrehoztam új jegyeket a WordNet és a szóbeágyazás módszerének segítségével. A WordNet módszerét jegyként előttem Luong és társai [90] alkalmazták szószintű kiértékelésre. Kutatásukban a poliszémia jelenségének kezelésére használták. Mondatszintű minőségbecsléshez a WordNetet jegyként más még nem használta előttem. Végül, a neurális módszerek kutatását is elkezdtem angol-magyar nyelvpárra.

5.3. Minőségbecslő rendszer angol-magyar nyelvpárra

Létrehoztam egy minőségbecslő rendszert angol-magyar nyelvpárra, amelyet Hun-QuEst-nek (Hungarian QuEst) neveztem el. A rendszer tanításához a HuQ korpuszt használtam. Mivel a HuQ korpusz különböző gépi fordítórendszer fordításait tartalmazza, amelyeknek a működése (pl. Google és Bing fordítók) nem hozzáférhető, ezért kutatásomban csak „black-box", azaz gépi fordítórendszertől független jegyeket használok.

Első feladatom az volt, hogy a meglévő QuEst rendszert alkalmaztam angol-magyar nyelvpárra. A QuEst alapértelmezetten 79 „black-box" jegyet tartalmaz[2] (tartalmazza a 17 alapjegykészletet), amelynek egy része nyelvspecifikus és nyelvi elemzőket használ (pl. Stanford parser, Berkeley Parser stb.).

Második feladatom az volt, hogy a QuEst keretrendszerbe magyar nyelvi elemzőket integráltam. A szófaji elemzéshez és egyértelműsítéshez a PurePos 2.0 [80] elemzőt használtam. A PurePos egy nyílt forráskódú, rejtett Markov-modellen alaplú morfológiai egyértelműsítő, amely a Humor [81] magyar morfológiai elemzőt is segítségül hívja. Jelenleg ez az elérhető legjobb („state-of-the-art") morfológiai egyértelműsítő, magyar nyelvre. A főnévicsoport-felismerés (NP-chunking) problémájára a HunTag [82] szoftvert használtam, amelyet a Szeged Treebank [83] korpuszon tanítottam. A HunTag egy maximum entrópia Markov-modellen alapuló, szekvenciális elemző.

[2]https://github.com/ghpaetzold/questplusplus/blob/master/config/features/features_blackbox-79.xml

DOI:10.15774/PPKE.ITK.2019.003

A 79 jegyből csak 76 jegyet tudtam implementálni (a teljes 76 jegyből álló jegykészlet az A.4. függelék A.1. táblázatában található). Mivel a maradék 3 jegy „content word”-el kapcsolatos, amelyeket nem tudtam egyértelműen leképezni a magyar nyelvi elemzőkkel, ezért azokat kihagytam. A QuEst rendszerbe idővel más kutatók által létrehozott további jegyeket is integráltak, de mivel azok olyan további nyelvspecifikus elemzőket igényelnek, amelyek nem álltak rendelkezésemre magyar nyelvre, ezért azokat sem implementáltam.

Kutatásomban először megvizsgáltam a 17 alapjegykészletet (a teljes 17 jegyből álló alapjegykészlet az A.4. függelék A.2. táblázatában található) magyar nyelvre, majd a 76 Specia és társai által létrehozott jegyet vizsgáltam meg, végül hozzáadtam az általam létrehozott új szemantikai jegyeket.

5.3.1. Szemantikai jegyek

Létrehoztam 75 darab új mondatszintű szemantikai jegyet (a teljes 75 jegyből álló jegykészletet az A.4. függelék A.3. táblázatában található). Célom az volt, hogy megvizsgáljam, a jelentés szempontjából, a forrásmondat és a gépi fordítás közötti hasonlóságot. A feladathoz szózsákokat (bag of words) hoztam létre, mind a forrásmondatból, mind a gépi fordításból. Egy szózsákban azonos szófajú szavak szótövei szerepelnek, a hozzájuk tartozó szinonimák és a szóbeágyazással kiszámolt szomszédaik. Végül az így elkészített szózsákok segítségével hoztam létre jegyeket.

Az első 3 jegy angol-magyar szótárat használ. A szótár a MetaMorpho szabály alapú gépi fordítórendszer által használt szótár [68], amely 365000 szópárt tartalmaz. A szótárban csak főnevek, igék és melléknevek vannak.

Minden forrásmondatra ($S = s_1, s_2, \ldots, s_i, \ldots, s_n$) és a hozzá tartozó gépi fordításra ($T = t_1, t_2, \ldots, t_j, \ldots, t_m$) megszámoltam, hogy hány lefordított szópár ($s_i; t_j$) található a szótárban. Az alábbi képleteket alkalmaztam:

$$SzótáriSzámolás_S = \frac{darab((s_i; t_j) \in D)}{n} \tag{5.1}$$

$$SzótáriSzámolás_T = \frac{darab((s_i; t_j) \in D)}{m} \tag{5.2}$$

DOI:10.15774/PPKE.ITK.2019.003

$$SzótáriSzámolás_S \text{ (5.1) és } SzótáriSzámolás_T \text{ (5.2) } harmonikus\ átlaga \tag{5.3}$$

, ahol D a szótár, n a forrásmondat hossza, m a gépi fordítás hossza, $i = [1, n]$ és $j = [1, m]$.

Továbbá létrehoztam 72 (3x24) darab jegyet a WordNet, az LSA és a szóbeágyazási modell segítségével. Az angol nyelvű WordNethez a Princeton WordNet 3.0-t [41] használtam, míg magyar nyelvre a Miháltz és társai által fejlesztett Hungarian WordNetet [39].

A szemantikai mérőszámok előállításához a WordNetekből először kigyűjtöttem a forrásmondat és a hozzátartozó gépi fordítás szavainak szinonima azonosítóit (synset ids). Ezek az azonosítók állnak az 1. szinten. Ezután a kigyűjtött szavakhoz hozzávettem a hipernimáját (2. szint) és azok hipernimáját (3. szint) is. Az így kigyűjtött szinonima és hipernima azonosítókból létrehoztam egy, a forrásmondathoz tartozó azonosító halmazt ($HALMAZ_S$), és egy, a gépi fordításhoz tartozó, azonosító halmazt ($HALMAZ_T$). Ezt követően kiszámoltam a létrehozott halmazok súlyozott metszetét (W): $I(S;T) = SET_S \cap SET_T = \{y_1, \ldots, y_k\}$. A szemantikai mérőszámokat az alábbi képletekkel számoltam ki:

$$WordNetSzámolás_S = \frac{W(I(S;T))}{n} \tag{5.4}$$

$$WordNetSzámolás_{x_S} = \frac{W(I(S;T))}{|x_S|} \tag{5.5}$$

$$WordNetSzámolás_T = \frac{W(I(S;T))}{m} \tag{5.6}$$

$$WordNetSzámolás_{x_T} = \frac{W(I(S;T))}{|x_T|} \tag{5.7}$$

$$WordNetSzámolás_S \text{ (5.4) és } WordNetSzámolás_T \text{ (5.6) } harmonikus\ átlaga \tag{5.8}$$

DOI:10.15774/PPKE.ITK.2019.003

$$WordNetSzámolás_{x_S}\ (5.5)\ és\ WordNetSzámolás_{x_T}\ (5.7)\ harmonikus\ átlaga \quad (5.9)$$

, ahol $|x|$ a főnevek száma a mondatban; n a forrásmondat hossza; m a gépi fordítás hossza; és

$$W(I(S;T)) = \sum_{i=1}^{k} \frac{y_i}{szint_{y_i}};\ y_i \in I(S;T)$$

A fenti 6 képlettel kiszámoltam még az igékre, melléknevekre és határozószókra is a szemantikai mérőszámokat, így összesen 24 darab új jegyet hoztam létre. A melléknevek és a határozószók esetében, mivel nincsen hipernima, a két fogalom közötti hasonlóságot kifejező relációkat használtam (similar_to és eq_near_synonym).

Abban az esetben, amikor a WordNet nem adott találatot – ami a magyar WordNet méretének köszönhetően gyakori jelenség volt – LSA vagy szóbeágyazási modell segítségével bővítettem a keresést.

Egyik kísérletemben a Siklósi és társai [84] által készített szóbeágyazási modellt alkalmaztam. Amikor nem adott eredményt a WordNet, a szóbeágyazási modell segítségével lekértem az adott szóhoz szemantikailag legközelebb álló 10 szomszédot, és a WordNet jegyek képleteivel azokra is kiszámoltam a mérőszámokat. Mivel a szóbeágyazási modell által kiadott eredmények nem szinonimák, ezért ezekben az esetekben a súlyt lecsökkentettem 0,1-re. Így létrehoztam még 24 WordNet jegyet, amelyek szóbeágyazást használnak (WordNet+WE).

Egy másik kutatásomban, amikor a WordNet használatával nem jutottam eredményre, azzal kísérleteztem, hogy az LSA módszerével kerestem egyezést. Itt csak egy darab egyezést kerestem, majd a kapott eredményre újra kiszámoltam a WordNet képleteivel a mérőszámokat, szintén 0,1-es súllyal. Így létrehoztam újabb 24 WordNet jegyet, az LSA felhasználásával (WordNet+LSA).

Ilyen módon a 24 darab alapértelmezett WordNet jegy, a 24 darab WordNet+WE jegy és a 24 darab WordNet+LSA jegy összesen kiadja a 72 darab WordNet jegyet.

DOI:10.15774/PPKE.ITK.2019.003

5.4. Mérések

Először megvizsgáltam, hogy a 17 alapjegykészlet (17F) hogyan teljesít angol-magyar nyelvpárra, majd a 76 Specia és társai által készített jegykészletet (76F) mértem le angol-magyar nyelvpárra. Végül hozzáadtam az általam készített szemantikai jegyeket is a modellhez. Megvizsgáltam, hogy a háromféle WordNet jegytípus (WordNet, WordNet+WE, WordNet+LSA) közül melyikkel értem el a legjobb eredményt, ezt követően azzal a jegytípussal végeztem el a többi mérést.

Mivel a három jegytípus közül csak a legjobb eredményt elérő típussal mértem tovább, ezért a további mérésekhez összesen 103 (76F + 3 szótári jegy + 24 WordNet jegy) jegyet használtam (103F). A különböző jegykészlet beállításokkal különböző minőségbecslő modelleket készítettem. A jegyek segítségével külön betanítottam egy-egy minőségbecslő modellt a TA, a GA, a TG, a CLTA, a CLGA és a CLTG értékeire.

Továbbá megvizsgáltam, hogy angol-magyar nyelvpárra melyik gépi tanuló algoritmus teljesít a legjobban. Kipróbáltam a lineáris regressziót, a Specia és társai kutatásában [58] használt Gauss-eljárást, a döntési fákat, a véletlen erdőt, a szupport vektor regressziót és a szupport vektor gépeket.

A döntési fák esetében a J48 (batch:100, confidence factor: 0.25) [85] algoritmust használtam, az SVM (RBF kernel, gamma:0,01, cache: 250007, epsilon: 1,0E-12, c: 1,0, batch: 100) és SVR esetében RBF kernelt [34] (gamma: 0,01, cache: 250007, c: 1,0, batch: 100). A kiértékeléshez 10-szeres keresztvalidációt használtam.

Végül végeztem jegykiválasztást is: Beck és társai kutatásai [76] alapján a 103 jegyből kiválasztottam a releváns jegyeket. Egyes jegyek ugyanis javítják a rendszer minőségét, de lehetnek olyan jegyek is, amelyek rontják azt. Ezért kiválasztottam azokat a releváns jegyeket, amelyek javítják a rendszer minőségét.

A jegyek kiválasztásához a korreláció alapú jegykiválasztó módszert, a döntési fa által nyújtott rangsort és az előrehaladó kiválasztás (forward selection) metódust is kipróbáltam.

Az optimalizált jegykészletek az alábbiak:

- OptTA: Optimalizált jegykészlet a TA értékekhez.
- OptGA: Optimalizált jegykészlet a GA értékekhez.

DOI:10.15774/PPKE.ITK.2019.003

- OptTG: Optimalizált jegykészlet a TG értékekhez.
- OptCLTA: Optimalizált jegykészlet a CLTA értékekhez.
- OptCLGA: Optimalizált jegykészlet a CLGA értékekhez.
- OptCLTG: Optimalizált jegykészlet a CLTG értékekhez.

5.5. Eredmények

Az eredmények táblázataiban (5.1., 5.2., 5.3., 5.6., 5.7., 5.10., 5.11. és 5.12. táblázat), azon eseteket, amelyeknél a magasabb érték a jobb eredmény, a ↑ jelöli, míg azokat az eseteket, ahol a kisebb érték a jobb, a ↓ jelöli.

Az első mérésem az volt, hogy kiválasszam, hogy melyik típusú WordNet jegyet használjam. Az 5.1. táblázatban láthatóak a WordNet kísérlet eredményei. Látható, hogy a szóbeágyazást használó WordNet jegyek érték el a legjobb eredményt, ezért a kutatás további részeiben a WordNet+WE jegyeket használtam.

	Korreláció ↑	MAE ↓	RMSE ↓
TG-17F (alapjegykészlet)	0,4931	0,8345	1,0848
TG-103F (WordNet)	0,5078	0,9304	1,1776
TG-103F (WordNet+LSI)	0,5347	0,8216	1,0507
TG-103F (WordNet+WE)	**0,5618**	**0,7962**	**1,0252**

5.1. táblázat A három típusú WordNet jegyek kiértékelése

Az 5.2., az 5.4. és az 5.3. táblázatban a tanuló algoritmusokkal való kísérletek eredményeit mutatom be. Az eredeti kutatásaimban a szupport vektor regresszió és a szupport vektor gépek érték el a legjobb eredményeket, ezért a disszertációmban leírt kutatásokban az SVR és az SVM módszereket használtam.

Azóta az ensemble módszerekkel további eredményjavulást tudtam elérni, ezek láthatóak az alsó részekben. Kivétel ez alól a bináris osztályozós feladat(lásd 5.4. táblázat): ott a szupport vektor gépek teljesítettek jobban.

Az 5.6., az 5.7. és az 5.8. táblázatban az angol-magyar minőségbecslés méréseinek eredményeit mutatom be. Látható, hogy a 17 jegyből álló nyelvfüggetlen alapjegykészlet teljesített a leggyengébben. A TG értékeire az alapjegykészlet még az 50%-os korrelációt

DOI:10.15774/PPKE.ITK.2019.003

	Korreláció ↑	MAE ↓	RMSE ↓
Lineáris regresszió	0,5347	0,8378	1,0343
Gaussi eljárás	0,5357	0,8366	1,0307
Véletlen erdő	0,556	0,8317	1,0277
Szupport vektor regresszió	**0,5618**	**0,7962**	**1,0252**
Bagging (véletlen erdő)	0,5677	0,8036	1,0051

5.2. táblázat Tesztelt algoritmusok regresszióra

	CCI ↑	MAE ↓	RMSE ↓
Döntési fa	55,2667%	0,3430	0,5022
Véletlen erdő	59,1333%	0,3557	0,4414
Szupport vektor gépek	**60,3333%**	**0,3347**	**0,4318**
Bagging (véletlen erdő)	60,6667%	0,3208	0,4012
Boosting (véletlen erdő)	61,1333%	0,2605	0,5018

5.3. táblázat Tesztelt algoritmusok osztályozásra (3 osztályattribútumos)

	CCI ↑	MAE ↓	RMSE ↓
Döntési fa	65,8%	0,3825	0,5084
Véletlen erdő	66%	0,3793	0,5550
Boosting (véletlen erdő)	66%	0,3423	0.5755
Bagging (véletlen erdő)	66,2%	0,3978	0,4534
Szupport vektor gépek	**67,8667%**	**0,3213**	**0,5669**

5.4. táblázat Tesztelt algoritmusok bináris osztályozásra

sem érte el, ami azt jelenti, hogy a 17 jegy gyengén függ össze az emberi kiértékeléssel. Ez adja a létjogosultságát annak a kutatásnak, amely során angol-magyar nyelvpárra releváns jegyeket kutattam.

A 5.5. táblázatban mutatom be a jegykiválasztó módszerek összehasonlítását. Az eredmények alapján a „forward selection” módszere nyújtotta a legjobb eredményt, ezért a kutatásom további részeiben ezt a módszert használtam.

	Korreláció ↑	MAE ↓	RMSE ↓
CFS (TG - 47 jegy)	0,5221	0,8248	1,0599
Döntési fa (TG - 86 jegy)	0,5537	0,7903	1,0336
Forward selection (TG - 26 jegy)	**0,6100**	**0,7459**	**0,9775**

5.5. táblázat Jegykiválasztó módszerek összehasonlítása

DOI:10.15774/PPKE.ITK.2019.003

Az eredményekből (lásd 5.6., 5.7. és 5.8. táblázat) továbbá az is látható, hogy a Specia és társai által fejlesztett további jegyek javítják a rendszer minőségét, ám amikor hozzáadtam a szemantikai jegyeket, további 1-2%-os eredményjavulást értem el.

	Korreláció ↑	MAE ↓	RMSE ↓
TA-17F (alapjegykészlet)	0,3832	0,9429	1,1990
TA-76F	0,4757	0,8804	1,1274
TA-103F	0,4847	0,8805	1,1199
OptTA (29 jegy)	**0,5245**	**0,8397**	**1,0869**
GA-17F (alapjegykészlet)	0,5400	0,8229	1,1278
GA-76F	0,5980	0,7751	1,0391
GA-103F	0,6070	0,7723	1,0297
OptGA (32 jegy)	**0,6413**	**0,7440**	**0,9878**
TG-17F (alapjegykészlet)	0,4931	0,8345	1,0848
TG-76F	0,5510	0,7984	1,0342
TG-103F	0,5618	0,7962	1,0252
OptTG (26 jegy)	**0,6100**	**0,7459**	**0,9775**

5.6. táblázat Hun-QuEst regressziós modelleinek kiértékelése

Az 5.1. táblázatban található érték és az 5.6. táblázatban lévő TG-76F érték összevetésével, az látható, hogy mind a WordNet, mind a WordNet+LSA jegyek rontottak az eredményen. Ebből arra következtetek, hogy a szóbeágyazás módszerének integrálásával sikerült elérni az eredményjavulást.

	CCI ↑	MAE ↓	RMSE ↓
CLTA-17F (alapjegykészlet)	54,9333%	0,3590	0,4591
CLTA-76F	57,1333%	0,3496	0,4478
CLTA-103F	57,6667%	0,3492	0,4483
OptCLTA (21 jegy)	**60,9333%**	**0,3370**	**0,4346**
CLGA-17F (alapjegykészlet)	58,8667%	0,3434	0,4419
CLGA-76F	62,1333%	0,3339	0,4301
CLGA-103F	62,4667%	0,3310	0,4275
OptCLGA (10 jegy)	**64,0667%**	**0,3299**	**0,4262**
CLTG-17F (alapjegykészlet)	57,8000%	0,3433	0,4417
CLTG-76F	60,0667%	0,3354	0,4327
CLTG-103F	60,3333%	0,3347	0,5495
OptCLTG (12 jegy)	**61,8000%**	**0,3299**	**0,4263**

5.7. táblázat Hun-QuEst 3 osztályattribútumos osztályozási modelleinek kiértékelése

DOI:10.15774/PPKE.ITK.2019.003

	CCI ↑	MAE ↓	RMSE ↓
CLBITA-17F (alapjegykészlet)	66,0000%	0,3400	0,5831
CLBITA-76F	66,4000%	0,3360	0,5797
CLBITA-103F	67,7333%	0,3227	0,5680
OptCLTA (4 jegy)	**68,0667%**	**0,3193**	**0,5651**
CLBIGA-17F (alapjegykészlet)	69,4667%	0,3053	0,5526
CLBIGA-76F	71,9333%	0,2807	0,5298
CLBIGA-103F	72,1333%	0,2787	0,5279
OptCLBIGA (13 jegy)	**72,6667%**	**0,2733**	**0,5228**
CLBITG-17F (alapjegykészlet)	65,7333%	0,3427	0,5854
CLBITG-76F	68,5333%	0,3147	0,561
CLBITG-103F	69,7333%	0,3027	0,5502
OptCLBITG (16 jegy)	**70,1333%**	**0,2987**	**0,5465**

5.8. táblázat A Hun-QuEst bináris osztályozási modelleinek kiértékelése

Az igazi eredménynövekedést a jegkiválasztás után értem el. A regressziós modelleknél ∼10%-os, míg az osztályozási modelleknél ∼5%-os eredményjavulást értem el az alapjegyhez képest. Emellett a regressziós modelleknél csupán a jegyek ∼30%-ával sikerült ezt a javulást elérni, míg az osztályozási modelleknél a jegyek ∼10-20%-ával. Ez azt is jelenti, hogy kevesebb erőforrásból, kevesebb futási idővel értem el magasabb eredményt.

A részletes eredmények az alábbiak:

- Az OptTA **29 jeggyel ∼14%-al** magasabb korrelációt ért el az alapjegykészlethez képest.
- Az OptGA **32 jeggyel ∼10%-al** magasabb korrelációt ért el az alapjegykészlethez képest.
- Az OptTG **26 jeggyel ∼12%-al** magasabb korrelációt ért el az alapjegykészlethez képest.
- Az OptCLTA **21 jeggyel ∼6%-al** több egyedet osztályozott helyesen az alapjegykészlethez képest.
- Az OptCLGA **10 jeggyel ∼5%-al** több egyedet osztályozott helyesen az alapjegykészlethez képest.

DOI:10.15774/PPKE.ITK.2019.003

- Az OptCLTG **12 jeggyel ∼4%-al** több egyedet osztályozott helyesen az alapjegykészlethez képest.
- Az OptCLBITA **4 jeggyel ∼4%-al** több egyedet osztályozott helyesen az alapjegykészlethez képest.
- Az OptCLBIGA **13 jeggyel ∼3%-al** több egyedet osztályozott helyesen az alapjegykészlethez képest.
- Az OptCLBITG **16 jeggyel ∼5%-al** több egyedet osztályozott helyesen az alapjegykészlethez képest.

Az összes optimalizált jegykészlet az A.4. függelék A.4., A.5., A.6., A.7., A.8. és A.9. táblazatában található.

Az optimalizált jegyek a relevancia sorrendjében. A vastagon kiemelt azonosítók az általam létrehozott szemantikai jegyek:

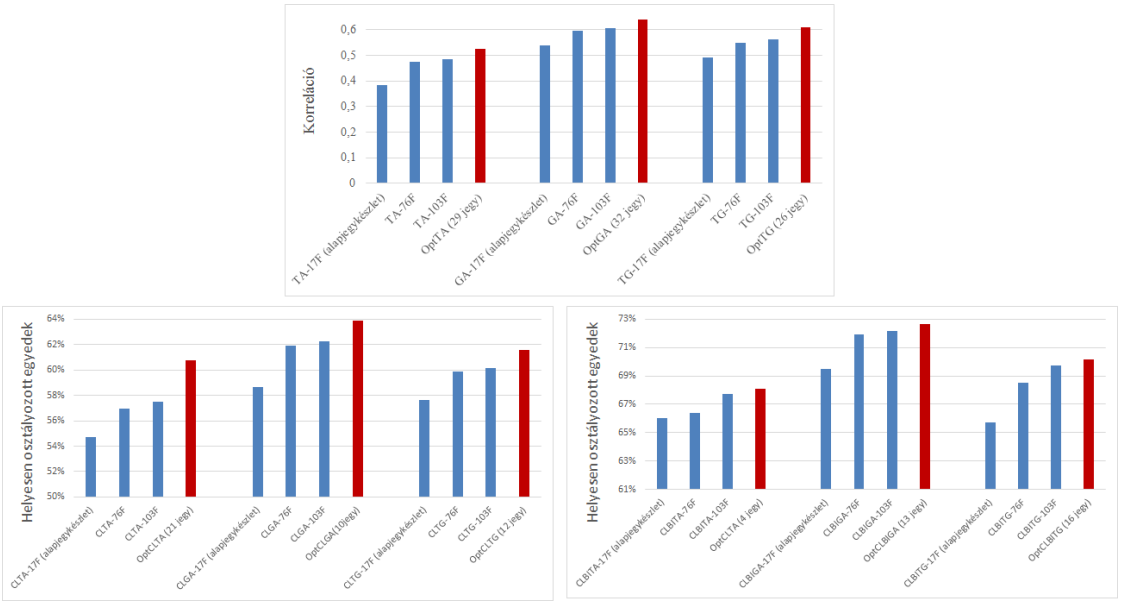
- OptTA 29 jegye: 1064, 1015, 1091, 1089, 2005, 1001, 1075, 1072, 1057, 1066, 1024, 1082, 1042, 1094, 1010, 1068, **2019**, 1006, 1060, 1013, **2023**, 1073, 1076, 1067, **2015**, **2029**, 1038, **2007**
- OptGA 32 jegye: 1015, 1060, 1002, 1082, 1091, **2019**, 1066, **2003**, 1036, 1068, 1072, **2020**, **2026**, 1006, 1010, 1089, 1044, 1073, 1054, 1046, 1093, 2005, **2007**, **2016**, 1067, 1011, 1052, **2001**, 1034, 1042, **2002**, **2015**
- OptTG 26 jegye: 1015, 1091, 1089, 1002, 1082, 1066, 1044, 1057, 1016, 1010, 1072, **2019**, 1006, 1068, 2005, **2001**, 1080, **2028**, 1013, 1052, **2022**, 1073, 1077, **2006**, 1067, 1079
- OptCLTA 21 jegye: 1068, 1064, 1005, 1091, 1092, 1015, **2001**, 1072, 1046, 1077, 1078, 1055, 1082, 1066, 1093, 1057, 1081, **2019**, 1067, 1090, 1010
- OptCLGA 10 jegye: 1064, 1076, **2002**, 1091, 1072, 1047, 1077, 1011, 1014, 1054
- OptCLTG 12 jegye: 1064, 1091, 1075, 1093, 1057, 1072, **2010**, **2025**, 1066, 1014, 1067, 1079
- OptCLBITA 4 jegye: **2029**, 2017, 1066, 1048

DOI:10.15774/PPKE.ITK.2019.003

- OptCLBIGA 13 jegye: **2021**, 1009, 1015, 1002, 1064, 1068, 1093, **2004**, 1001, **2022**, 1072, 1078, 1011
- OptCLBITG 16 jegye: 1015, 1060, 1066, 1072, 1034, 1010, 1090, **2012**, **2019**, 1075, 1051, 1078, **2005**, 1068, 1055, 1073

Az optimalizált jegykészleteket vizsgálva az látható, hogy mindegyik halmazban található releváns szemantikai jegy, és az esetek többségében több ilyen jegy is található.

A könnyebb átláthatóság végett diagrammon ábrázoltam az általam készített modellek kiértékeléseit (lásd 5.1. ábra). A regressziós modellek esetében a korreláció értékeket, míg az osztályozós modellek esetében a helyesen osztályozott egyedek számát ábrázoltam. Látható, hogy az optimalizált jegyhalmazok (pirosan jelölt oszlop) minden esetben a legjobb eredményt érték el.



5.1. ábra A Hun-Quest modelljeinek kiértékelése

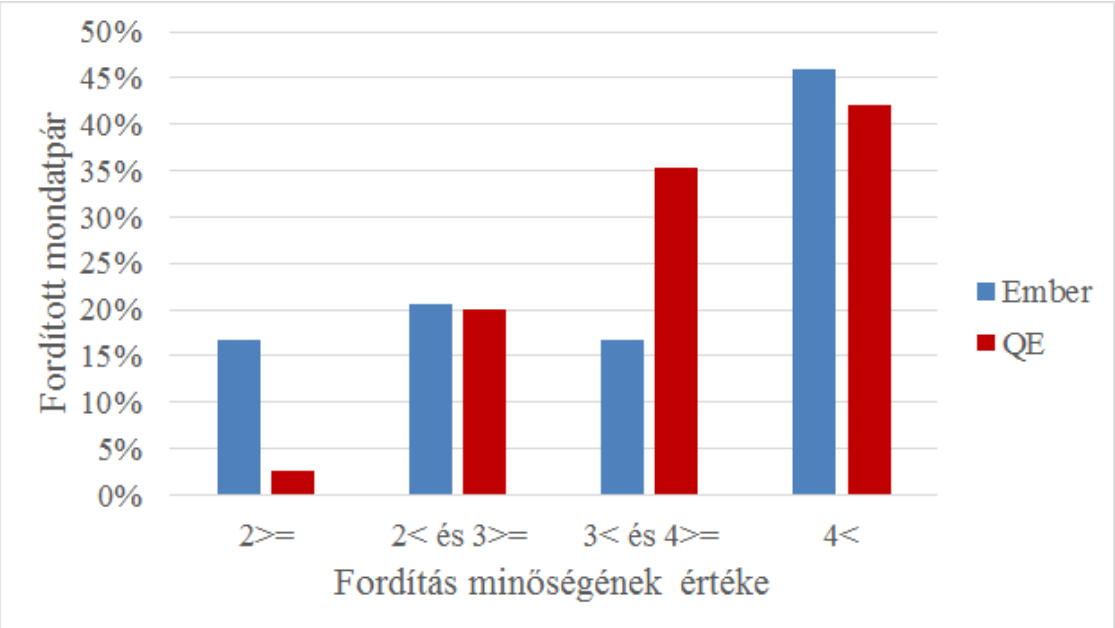
Az 5.9. táblázatban látható néhány példában az „FM” a forrásmondatot jelöli. A modell az első fordítás esetében produkálta a legrosszabb becslést: ∼3 értéket rontott. A minőségbecslő modell azért is adhatott ilyen jó értéket, mert több szót is jól lefordított a gép, csak mivel mondatszinten értelmetlen a mondat, ezért lehetett alacsony az emberi kiértékelés. A második példa hasonló az elsőhöz, csak egy kicsit olvashatóbb. A harmadik példa egy tökéletes fordítás, de valószínűleg a minőségbecslő modell nem

DOI:10.15774/PPKE.ITK.2019.003

QE	Ember	
4,166	1,167	FM: Necessary, however, is the evil; necessary are the envy and the distrust and the back-biting among the virtues. MT: A gonosz, szükséges az irigység és a bizalmatlanság és a back-biting a.
4,841	2,667	FM: Civilization should flow with milk and honey for you. MT: Civilization kell folynia a tejjel és mézzel az Ön számára.
3,974	5	FM: The florida keys! MT: A Floridai kulcsok!
4,832	4,833	FM: Andy, I want you to stay away from the rocks. MT: Andy, azt akarom, hogy ne menj a sziklákhoz.
4,999	5	FM: That's not good news. MT: Ez nem jó hír.

5.9. táblázat Néhány példa

ismerte a „Florida" szót, ezért gyengébb minőséget adott neki. Az utolsó kettő példában majdnem megegyezik a modellem által becsült érték az emberi kiértékeléssel. Az utolsó előtti példa azért érdekes, mert érezzük a finom jelentésbeli különbséget a forrásmondat és a gépi fordítás között. A gép ebben az esetben is az emberi kiértékeléshez igen közeli értéket adott.



5.2. ábra A minőségbecslő modell összehasonlítása az emberi kiértékeléssel

DOI:10.15774/PPKE.ITK.2019.003

Az 5.2. diagrammon látható, hogy a minőségbecslő modell inkább pozitívabban értékel. Nagyon kevés a 2, vagy annál kisebb érték. A 3, vagy annál jobb minőségű fordítások száma ∼77%-ot tesznek ki. Ez azért is lehet, mert a HuQ korpusz sokkal több 4, vagy annál magasabb értékű fordítást tartalmaz, ezért a minőségbecslő modell tanítása eltolódott a magasabb minőségű fordítások felé. Feltételezem, hogy ha a modell jobban rátanul a rosszabb minőségű fordításokra, a korreláció mértéke is magasabb lesz.

5.6. A WordNet jegyek kiterjesztése más nyelvpárokra

Az általam létrehozott WordNet jegyeket megvizsgáltam angol-spanyol és angol-német nyelvpárokra is. Mivel az angol-magyar nyelvpárra végzett kutatásom azt mutatta, hogy a WordNet jegyek voltak az inkább releváns jegyek a szótári jegyekkel szemben, ezért a WordNet jegyeket implementáltam angol-spanyol és angol-német nyelvpárokra. Amikor a kutatásomat végeztem, német és spanyol nyelvre még nem állt rendelkezésre szóbeágyazási modell, ezért a WordNet jegyek szóbeágyazási modell szerinti részeit nem implementáltam.

A kutatáshoz az „ACL 2014 Ninth Workshop on Statistical Machine Translation”[3] versenyen biztosított korpuszokat használtam. Mivel a jegyeim mondatszintűek, ezért a megosztott feladatokból három mondatszintű feladatot oldottam meg:

1. „Érzékelt utómunka ráfordítás” (Perceived Post-Editing effort - PPEE) becslése: A tanítókorpusz (C1a) 3816 angol-spanyol kiértékelt mondatpárt tartalmaz, míg a tesztkorpusz 600 kiértékelt mondatpárt.

2. „Érzékelt utómunka ráfordítás” (Perceived Post-Editing effort - PPEE) becslése: A tanítókorpusz (C1b) 1400 angol-német kiértékelt mondatpárt tartalmaz, míg a tesztkorpusz 600 kiértékelt mondatpárt.

3. „Szükséges javítások aránya” (Percentage of Edits Needed - HTER) becslése: A tanítókorpusz (C2) 896 angol-spanyol kiértékelt mondatpárt tartalmaz, míg a tesztkorpusz 208 kiértékelt mondatpárt.

[3] http://www.statmt.org/wmt14/quality-estimation-task.html

DOI:10.15774/PPKE.ITK.2019.003

	Korreláció ↑	MAE ↓	RMSE ↓
HTER (C2) - alapjegykészlet	0,4078	0,1444	0,2117
HTER (C2) - alapjegykészlet + WordNet jegyek	**0,4149**	**0,1438**	**0,2106**
PET (C3) - alapjegykészlet	0,6677	0,1527	0,2246
PET (C3) - alapjegykészlet + WordNet jegyek	**0,6715**	**0,1522**	**0,2235**

5.10. táblázat C2 és C3 kiértékelése

	CCI ↑	MAE ↓	RMSE ↓
PPEE (C1a) - alapjegykészlet	58,67%	0,3450	0,4437
PPEE (C1a) - alapjegykészlet + WordNet jegyek	**59,46%**	**0,3420**	**0,4403**
PPEE (C1b) - alapjegykészlet	51,00%	0,3863	0,4880
PPEE (C1b) - alapjegykészlet + WordNet jegyek	**52,43%**	**0,3684**	**0,4693**

5.11. táblázat C1a és C1b kiértékelése

4. „Utómunkára ráfordított idő” (Post-Editing Time - PET) becslése: A tanítókorpusz (C3) 650 angol-spanyol kiértékelt mondatpárt tartalmaz, míg a tesztkorpusz 208 kiértékelt mondatpárt.

A spanyol nyelvű szinonimák kinyeréséhez az MCR 3.0 (Multilingual Central Repository 3.0) WordNetet [86] használtam, míg a német nyelvű szinonimák kinyeréséhez az UWN-t (Universal Multilingual Wordnet) [87] használtam.

Az 5.10. és az 5.11. táblázatban láthatóak a kutatás eredményei. Az alapjegykészletet eredetileg angol-spanyol nyelvpárra optimalizálták, ezért várható volt, hogy az általam készített minőségbecslő modell nem fog nagymértékű eredményjavulást produkálni. Továbbá a magyar, a spanyol és a német WordNet mérete is igen kicsi. A spanyol WordNetben nincsenek határozószók, és igéből is csak 37 darab található. Azonban ezek ellenére is sikerült ∼1%-os javulást elérni, a szemantikai jegyek hozzáadásával. A német korpusszal is sikerült több mint 1%-os javulást elérni. A WordNetek méretének növelésével, valamint szóbeágyazási modell integrálásával a mért eredmények további javulása is várható lenne. Azonban az eredményekből az is egyértelműen látszik, hogy az általam létrehozott jegyek kiterjeszthetőek más nyelpárokra.

DOI:10.15774/PPKE.ITK.2019.003

5.7. Neurális minőségbecslés angol-magyar nyelvpárra

Amikor 2014-ben elkezdtem kutatásomat, még nem végeztek komoly kutatásokat a neurális hálózat alapú minőségbecslés területen. A neurális hálózat már elkezdte térhódítását, de még nem volt olyan jelentős, mint ma. Ezért a kutatásomban csak a jegyek előállításához használtam a neurális hálózatot (nyelvmodell készítése, szemantikai hasonlóság mérése stb.). Azóta viszont, látva a neurális hálózatok dominanciáját, az elmúlt fél évben az immár teljeskörűen neurális alapú minőségbecslés kutatását is elkezdtem. Ugyanakkor e rövid idő alatt még nem tudtam megfelelő eredményeket elérni.

Implementáltam angol-magyar nyelvre a POSTECH és a BI-RNN rendszereket. A modellek tanításához a deepQuest [60] keretrendszert használtam. A tanításhoz a Hunglish [67] korpuszt és a HuQ korpuszt (lásd 4. fejezet) használtam.

Az 5.12. táblázatban láthatóak az angol-magyar nyelvre betanított neurális minőségbecslő modellek kiértékelésének eredményei. Látható, hogy a POSTECH sokkal jobban teljesített, mint a BI-RNN. Ez azért is lehetséges, mert a POSTECH módszernek van egy előtanítási fázisa, amiben előre betanítunk egy nagy párhuzamos korpusszal egy modellt. A BI-RNN esetében nincsen előtanítási fázisa, ezért kutatásomban a HuQ korpuszból vett 1000 mondatpárral tanítottam csak a modellt, míg a POSTECH esetén a Hunglish korpusz körülbelül 1 millió mondatpárjával tanítottam azt. Azonban ezzel együtt is csak ∼38%-os korrelációt sikerült vele elérni.

	Korreláció ↑	MAE ↓	RMSE ↓
POSTECH	0,3844	2,8453	3,1086
BI-RNN	0,0242	2,4242	2,7238

5.12. táblázat Neurális minőségbecslő rendszer kiértékelése

5.8. Továbblépési lehetőségek

A jelen kutatások [91] egyértelműen a neurális hálózaton alapuló minőségbecslés irányába mutatnak. Ez irányú kutatásom még kezdeti fázisban van, ezért továbblépésként mindenképpen ez az irány lenne indokolt. Ugyanakkor több idő szükséges ahhoz, hogy értékelhető eredményeket érhessek el benne.

DOI:10.15774/PPKE.ITK.2019.003

További kutatási irány lehet még – angol-magyar nyelvre – a szószintű, a kifejezésszintű és a dokumentumszintű minőségbecslés.

Végül, a vállalatok számára, rendkívül fontos feladat megbecsülni a gépi fordítás javítására szánt emberi munka mennyiségét és idejét. Erre a törekvésre láthattunk példát a 5.6. fejezetben található más nyelvekre végzett kutatásomban, ahol az emberi kiértékelések a ráfordított utómunka mennyisége (Perceived Post-Editing effort), a szükséges javítások aránya (Percentage of Edits Needed) és az utómunkára ráfordított idő (Post-Editing Time) mértéki voltak. Ezeken a területeken angol-magyar nyelvpárra is érdemes lenne kutatásokat végezni.

5.9. Összegzés

Létrehoztam egy angol-magyar minőségbecslő rendszert, amit azelőtt más még nem készített.. A rendszer tanításához a HuQ korpuszt használtam. A felépített rendszeren különböző méréseket végeztem el. Először az angol-spanyol nyelvre optimalizált alapjegykészletet mértem le angol-magyar nyelvre, majd megvizsgáltam a Specia és társai által implementált 76 jegykészletet is. Ezt követően saját szemantikai jegyekkel kísérleteztem. A szemantikai jegyekhez angol-magyar szótárat, a WordNetet, a szóbeágyazási modellt és az LSA módszert használtam. Mivel az LSA módszer és a szótár is csak kis mértékben produkált eredményjavulást, ezért ezekbe az irányokba nem folytattam további kutatásokat. Ellenben a WordNet jegyekkel és a szóbeágyazási modellek segítségével sikerült a legjobb esetben ∼14%-os eredménynövekedést elérni. Végeztem optimalizálást is: kevesebb releváns jeggyel sikerült jobb eredményt elérni. A legmagasabb, ∼14%-os eredményjavulást csupán 29 jeggyel sikerült elérni, ami az összes jegy számosságának csupán ∼35%-a.

Továbbá kiterjesztettem a WordNet jegyeket angol-spanyol és angol-német nyelvpárra. Mivel az alapjegykészletet angol-spanyol nyelvpárra optimalizálták, és a rendelkezésemre álló WordNetek mérete sem volt nagy, ezért nem is vártam nagy eredménynövekedést, ugyanakkor így is sikerült – minden esetben – ∼1%-os javulást produkálni. Angol-német nyelvpárra szintén ∼1%-os eredménynövekedést értem el.

DOI:10.15774/PPKE.ITK.2019.003

Kapcsolódó tézisek

2. tézis: **Létrehoztam 27 darab új szemantikai jegyet, kétnyelvű szótárral, a WordNet és a szóbeágyazás módszerével, amelyekkel eredményjavulást értem el az alapjegykészlethez képest.**

3. tézis: **Az általam létrehozott angol-magyar korpusz, valamint a 27 darab új szemantikai jegy segítségével, a QuEst keretrendszer alapján, magyar nyelvtechnológiai eszközök integrálásával, létrehoztam egy minőségbecslő rendszert, angol-magyar nyelvre.**

A tézishez kapcsolódó publikációk: [4] [6] [7] [9] [10].

DOI:10.15774/PPKE.ITK.2019.003

6. fejezet

A MaTros rendszer

6.1. Előzmények

Az elmúlt évtizedekben a gépi fordítórendszerek jelentős változásokon mentek keresztül. Mind a kutatók, mind a vállalatok törekednek a lehető legjobb fordítás előállítására. A cél elérésében alapvetően két irány határozza meg a kutatásokat. Az egyik esetben a kutatások arra irányulnak, hogy egy adott módszer hatékonyságát és minőségét növeljék, míg a másik esetben több különböző módszert kombinálnak a jobb eredmény reményében. Az egyes gépi fordítórendszereknek - eltérő viselkedésük miatt - megvannak a maguk előnyei és hátrányai. A gépi fordítórendszerek kombinációjával lehetőség nyílik a különböző módszerek előnyeinek egyesítésére, valamint a rendszerekben rejlő problémák gyengítésére. Így a létrehozott rendszer minőségében meghaladja a kombinációban részt vett rendszerekét.

A módszerek kombinálása történhet egyrészt a rendszerek ötvözésével (lásd 2.1.4. fejezet), másrészt a rendszerek kimeneteinek kombinálásával.

Kutatásom során, a minőségbecslés módszerét használtam különböző gépi fordítórendszerek kimenetének kombinálására. Célom az volt, hogy felhasználjam és teszteljem az általam létrehozott minőségbecslő rendszert a gyakorlatban. Létrehoztam egy minőségbecslő rendszert, amellyel kombináltam egy kifejezésalapú statisztikai gépi fordítórendszer, egy hierarchikus statisztikai gépi fordítórendszer és egy neurális gépi fordítórendszer kimenetét.

DOI:10.15774/PPKE.ITK.2019.003

6.2. Kapcsolódó munkák

A gépi fordítás kombinációja többféleképpen történhet. Történhet egyrészt a különböző rendszerek módszertanának kombinációjával (lásd 2.1.4. fejezet), például szabályalapú gépi fordítórendszerhez fordítómemóriát integrálni stb. Másrészt a kombináció úgy is létrejöhet, hogy a különböző gépi fordítórendszer kimeneteit, magukat a fordításokat kombináljuk. A kutatásomban ez utóbbira, a gépi fordítás kimenetének kombinálására fókuszáltam.

Huang és Papineni [100] egy hierarchikus rendszerkombinációt hoztak létre, ahol a rendszer igény szerint képes szó-, frázis- és mondatszintű kombinációra.

A leggyakoribb kombinációs módszerek helyettesítési gráf (confusion network) létrehozásával valósítják meg a rendszerek kimenetének egyesítését. A helyettesítési gráfot általában egy vázmondat köré építik [93, 101], amihez egynyelvű szóösszekötéssel kapcsolják a hipotéziseket. Az így létrehozott helyettesítési gráfalapú dekódoló segítségével választják ki a legvalószínűbb fordítást. Rosti és társai [102] a gépi fordítórendszerek kimenetéből hipotézisalapú helyettesítési gráfot építettek. Az egyik rendszer legjobb fordítását használták vázként, amit szószinten kiegészítettek a többi rendszerből származó alternatív fordításokkal. Ebből a hálózatból az általuk létrehozott dekódoló választotta ki a legjobb fordítást. Ezt Heafield és társai [103] úgy fejlesztették tovább, hogy a helyettesítési gráfot nem szószinten, hanem kifejezés szinten építették. Rosti és társai, valamint Heafield és társai a kombináláshoz szükséges számolásokhoz a TER algoritmust választották, míg Okita és társai [104, 94] a BLEU és a minőségbecslés módszereivel is kísérleteztek.

Kutatásomban egy az előzőektől független megközelítést választottam. Különböző gépi fordítórendszerek kimenetét kombináltam úgy, hogy magukat a fordításokat nem módosítottam. A minőségbecslés módszerével kiválasztottam a különböző gépi fordítórendszerek kimenetei közül a legjobb minőségű fordítást, és azt adtam a rendszer végső kimenetének.

DOI:10.15774/PPKE.ITK.2019.003

6.3. A felhasznált források és gépi fordítórendszerek bemutatása

Ennek a tanulmánynak a célja, hogy vállalati környezetben is kipróbáljam a kutatásom módszereit. Kísérleteimhez a MorphoLogic Lokalizáció Kft.[1] biztosította a korpuszokat, amelyek nagy része nem publikus.

6.3.1. Felhasznált gépi fordítórendszerek

A kutatásomat vállalati környezetben végeztem el, ezért a méréseket is a vállalati környezethez kellett igazítani. Bár közvetlenül nem férhetem hozzá a vállalat által használt gépi fordítórendszerekhez, a beállítási paramétereket viszont megkaptam.

A statisztikai gépi fordításhoz a Moses [69] keretrendszert használtam. A tanítóanyagot az előfeldolgozás műveleteinek vetettem alá (pl. tokenizálás, truecaseing stb.). A szóösszekötéshez a GIZA++ [95] rendszert, míg a nyelvmodell előállításához az IRSTLM [96] eszközt használtam. A felhasznált korpusz XML címkéket tartalmaz, amelyeket a m4loc[2] program segítségével kezeltem.

A neurális fordításhoz az OpenNMT [97] nevű szabadon elérhető keretrendszert használtam. Az OpenNMT keretrendszerben a gépi fordításhoz többféle enkóder-dekóder architektúrát implementáltak, valamint a szövegek előfeldolgozásához szükséges programok is elérhetőek. Munkám során az enkódoláshoz egy LSTM-alapú kétirányú RNN architektúrát használtam, míg dekódolónak egy „attention" modellt [98] alkalmaztam. Mindegyik rendszer 18 epochig tanult, valamint az SGD (Stochastic Gradient Descent) optimalizáció módszerét [99] használta.

A gépi fordítórendszerek tanításához használt korpuszok témái és méretei a 6.1. táblázatban szerepel.

A kutatásom során, a különböző nyelvekre, egy kifejezésalapú statisztikai gépi fordító (Phrase-Based Statistical Machines Translation - PBSMT), egy hierarchikus statisztikai gépi fordító (Hierarchical-Based Statistical Machine Translation - HBSMT) és egy neurálishálózat-alapú gépi fordító (Neural Machine Translation - NMT) egyikét használtam.

[1] http://www.morphologic-localisation.eu
[2] https://github.com/achimr/m4loc

DOI:10.15774/PPKE.ITK.2019.003

6.3.2. Felhasznált korpuszok

A kutatásomban négy különböző nyelvpárral kísérleteztem. Mind a négy nyelvpár esetén, a forrásnyelv az angol volt, míg a célnyelv a magyar, a német, az olasz és a japán volt. A kiválasztott négy nyelv strukturálisan rendkívül különböző, ez pedig jól reprezentálja a módszerem teljesítményét. A felhasznált korpuszok témájukat tekintve is különbözőek. A 6.1. táblázatban látható a különböző korpuszok témája és mérete. Az IT és az autóipari témájú korpuszokban a mondatok nagy része rövid, míg a jogi szövegek mondatai hosszúak és nyelvtanilag komplexek. Mivel a korpuszok nagy része nem volt publikus, ezért az újra implementálhatóság végett, az angol-német nyelvpárra a nyilvánosan elérhető *„The Acquis Communautaire multilingual parallel corpus"*[3]-t használtam.

Nyelvpár	Téma	MT tanítóanyag mérete	QE tanítóanyag mérete	QE tesztanyag mérete
angol - magyar (en-hu)	autóipar	240 000	6000	1500
angol - német (en-de)	jog	1 000 000	5250	1300
angol - olasz (en-it)	termékleírás	800 000	3143	785
angol - japán (en-ja)	IT	1 000 000	3169	790

6.1. táblázat Kutatáshoz használt korpuszok

A kutatásomban a minőségbecsléshez kapott korpuszokat felosztottam, 80-20% arányban, tanító- és tesztanyagra.

6.4. Mérések és eredmények

A kutatásomat a vállalati környezethez kellett igazítani, ezért nem használhattam emberi kiértékelést. A vállalat által használt kiértékelési módszereket kellett alapul vennem, ezért a minőségbecslő rendszerek tanításához automatikus mértékeket használtam: BLEU, OrthoBLEU és OrthoTER mértékeket. Ez azt eredményezte, hogy a minőségbecslő modelleim egy 0 és 1 közötti értéket becsültek attól függően, hogy melyik mértékre tanítottam be. Az OrthoBLEU (oBLEU) és OrthoTER (oTER) karakteralapú módszerek, amelyek a ragozó nyelvek esetén (pl. magyar) pontosabb becslést adnak a szóalapú

[3]https://ec.europa.eu/jrc/en/language-technologies/jrc-acquis

DOI:10.15774/PPKE.ITK.2019.003

BLEU módszerhez képest. A szóalapú BLEU módszer egyik hátránya ilyen esetekben, hogy ha egy szó csak toldalékban különbözik, de a szótő megegyezik, akkor két különböző szóként kezeli.

A minőségbecslő modell tanításához jegyekre volt szükség. Ehhez a feladathoz a QuEst [58] keretrendszert használtam, és kizárólag nyelvfüggetlen, gépi fordítórendszertől független jegyeket.

A kutatásomban 67 jegyet alkalmaztam (a teljes jegykészlet a B.1. függelék B.1. táblázatban található), melyeket Specia és társai [58] fejlesztettek. Ezek tartalomhűségre és nyelvhelyességre vonatkozó jegyeket is tartalmaznak (pl. tokenek száma a forrás és a célnyelvi mondatban, célnyelvi mondat perplexitása stb.).

Az angol-magyar minőségbecslés eredményei (lásd 5. fejezet) és az elmúlt évek kutatásai alapján [77, 78], a szupport vektor regresszió (RBF kernel, gamma: 0,01, cache: 250007, c: 1,0, batch: 100) bizonyult a legeredményesebb módszernek, ezért a kutatásomban ezt a módszert használom.

Magyar nyelvre végeztem egy kísérletet, amiben hozzáadtam további jegyeket (az eredmények szekcióban *en-hu+*-ként hivatkozom rá). A magyar nyelvre készült kutatásom (lásd 5. és 7. fejezet) alapján a 67 jegy mellett további 60 jegyet próbáltam ki (a teljes jegykészlet a B.1. függelék B.2. táblázatban található), melyekből 53 saját fejlesztésű. Ezek nyelvspecifikus (pl. igék és főnevek aránya a célnyelvi mondatban, igekötők aránya stb.), n-gram (pl. célnyelvi mondat perplexitása, célnyelvi mondat nyelvmodell valószínűsége stb.), hiba (pl. ismeretlen szavak aránya a célnyelvi mondatban, XML címkék aránya a mondatban stb.) és szemantikai (pl. WordNet jegyek, szótár jegyek stb.) jegyeket tartalmaznak.

Mivel a cél az volt, hogy minél jobb eredményt érjek el, ezért a különböző gépi fordítórendszerekhez külön minőségbecslő modellt tanítottam be (lásd 6.1. ábra).

A következő alfejezetben bemutatom az általam felépített kompozit fordítórendszert.

6.4.1. A kompozit gépi fordítórendszer

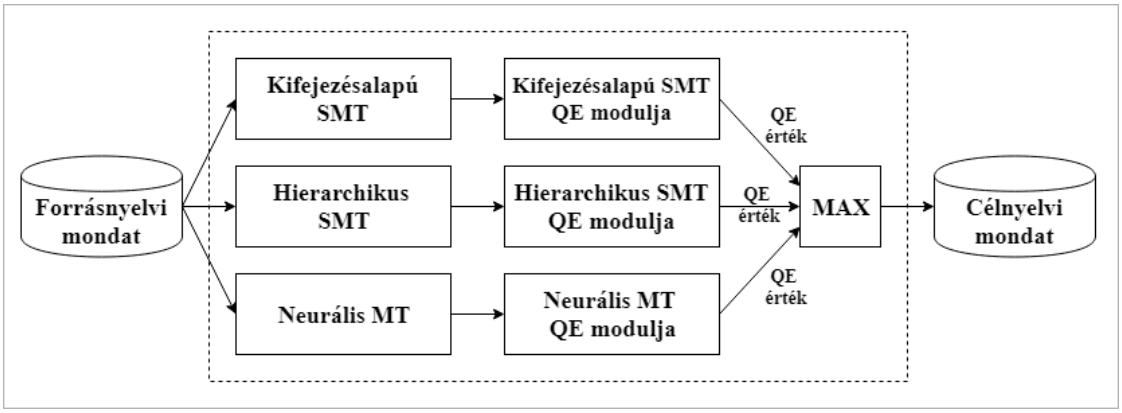
A kompozit gépi fordítórendszer (MaTros rendszer) a különböző gépi fordítórendszerek (PBSMT, HBSMT és NMT) kombinálásával jött létre. A rendszer architektúrája a 6.1. ábrán látható. A rendszer a különböző gépi fordítórendszerek segítségével feldol-

DOI:10.15774/PPKE.ITK.2019.003

gozza a forrásnyelvi bemenetet, majd a minőségbecslő modellek segítségével megbecsüli a lefordított mondatok minőségét. Ez alapján a rendszer kiválasztja azt fordítást, amelyhez a legmagasabb (6.1. ábrán a MAX jelöli) minőségi érték tartozik, és ez lesz a rendszer végső kimenete.

A tanítás során a három gépi fordítórendszerhez külön-külön betanítottam egy-egy minőségbecslő modellt.

Kutatásomban az angol-olasz és angol-japán nyelvpárokra csak a PBSMT és a HBSMT rendszereket kombináltam, míg az angol-német és angol-magyar nyelvpárok esetén a PBSMT, a HBSMT és az NMT rendszerek kimeneteit kombináltam.



6.1. ábra A kompozit gépi fordítórendszer architektúrája

Kísérletem során a tanító- és tesztanyagokat a különböző gépi fordítókkal lefordítottam. Ezt követően a minőségbecslő modellek a forrás- és a lefordított mondatokból kinyerték a minőségi mutatókat, és megbecsülték a fordítások minőségét. Majd a becsült minőségek alapján a kompozit rendszer kiválasztotta a forrásmondatokhoz tartozó legmagasabb minőségi értékű fordítást.

6.4.2. Eredmények

A kutatásomban 4 nyelvpárt és 3 különböző kiértékelési mértéket használtam. A 6.2. táblázatban láthatók a BLEU, OrthoBLEU és OrthoTER mértékekre betanított modellek által becsült értékek. Az eredményekben bemutatom a különböző gépi fordítórendszerekre betanított modellek és a kompozit rendszer (Composite Machine Translation - CoMT) teljesítményeit. Látható, hogy a kiértékelés során, az összes vizsgált esetben, az általam

DOI:10.15774/PPKE.ITK.2019.003

		en-hu	en-hu+	en-de	en-it	en-ja
BLEU átlag ↑	PBSMT	0,5156		0,6288	0,7513	0,5945
	HBSMT	0,6157		0,4808	0,6998	0,6044
	NMT	0,6281		0,4364	-	-
	CoMT	**0,6926**	**0,6978**	**0,6662**	**0,7525**	**0,6057**
	maxMT	0,7614		0,7330	0,7660	0,6458
oBLEU átlag ↑	PBSMT	0,7381		0,6757	0,8202	0,5361
	HBSMT	0,7679		0,6221	0,7993	0,5536
	NMT	0,7252		0,6751	-	-
	CoMT	**0,7729**	**0,7734**	**0,6855**	**0,8246**	**0,5553**
	maxMT	0,8698		0,7509	0,8374	0,5832
oTER átlag ↓	PBSMT	0,2903		0,3574	0,1669	0,4281
	HBSMT	0,2193		0,4170	0,1995	0,4075
	NMT	0,2101		0,2653	-	-
	CoMT	**0,1892**	**0,1871**	**0,2649**	**0,1662**	**0,4055**
	maxMT	0,0996		0,2083	0,1542	0,3769

6.2. táblázat Kombinált rendszerek kiértékelése

létrehozott kombinált rendszer jobb eredményt ért el, mint a vizsgált rendszerek önmagukban. A 6.2. táblázatban a ↑ nyíl azt jelöli, amikor a nagyobb érték a jobb eredmény, míg a ↓ nyíl azt, amikor a kisebb érték jelöli a jobb eredményt.

Az eredmények mélyebb vizsgálata során az látható, hogy az NMT rendszer minősége eltért az általam elvárt esettől. A neurális gépi fordítórendszer bevezetésével azt vártam, hogy a statisztikai rendszerek érvényüket vesztik. Igaz, hogy a neurális rendszer az esetek többségében a statisztikai rendszereknél átlagosan jobb minőséget eredményez, de a mondatszintű vizsgálat során, vannak olyan esetek, ahol a statisztikai rendszerek fordításai bizonyultak jobbnak. Az, hogy bizonyos mondatokat az SMT, míg másokat az NMT rendszer fordít jobban, megerősíti a fordítórendszer kombinációjának hasznosságát.

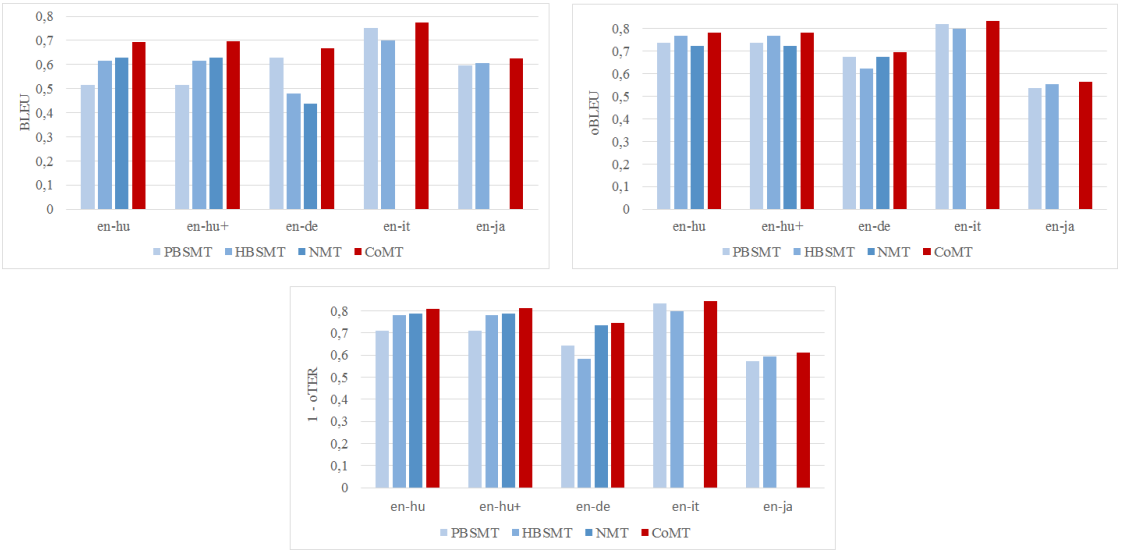
Felvetődött a kérdés, hogy mekkora lehetőség van további minőségnövekedés elérésére. Ennek kiderítésére úgy vizsgáltam meg az ideális becslő rendszer minőségét, hogy a teszthalmazon kiválasztottam a legjobb fordítási javaslatot az adott metrika alapján, mintha a minőségbecslő modelleim tökéletesen osztályoznának (ha minden esetben rendelkezésünkre állna referenciafordítás, és közvetlenül alkalmaznánk az automatikus metrikákat). Az így létrehozott ideális rendszer eredményei a 6.2. táblázatban a maxMT sorban olvashatóak. Láthatóan a maxMT rendszer minősége jelentősen jobb, mint az al-

DOI:10.15774/PPKE.ITK.2019.003

rendszereké külön-külön. Például mind a szóalapú, mind a karakteralapú BLEU esetén is egyaránt 5-15% javulás figyelhető meg. Ebből az következik, hogy sok mondat esetén van jelentős eltérés az SMT és az NMT rendszerek fordításai között.

Ennek ellenére a 6.2. táblázatból az is kiolvasható, hogy a kombinált rendszer eredménye messze elmarad a maxMT rendszer eredményeitől. Ehhez képest csak kisméretű javulást lehetett kimutatni a legjobb fordító alrendszerhez képest. Ebből az következik, hogy még jelentős tartalék rejlik a minőségbecslő rendszer becslési pontosságában.

A könnyebb átláthatóság végett diagrammon ábrázoltam az általam készített modellek kiértékeléseit (lásd 6.2. ábra).



6.2. ábra Kombinált rendszerek modelljeinek kiértékelése

Ezt követően megvizsgáltam, hogy az általam létrehozott jegyek hozzáadásával hogyan teljesítenek a modellek angol-magyar nyelvpárra. A modellek összehasonlítására és kiértékelésére az MAE, az RMSE és a Pearson-féle korreláció mértékeket használtam. A 6.3. táblázatban látható, hogy mindegyik modellnél sikerült javítani a modell minőségén. A 6.2. és a 6.3. táblázatokban arra mutatok rá, hogy angol-magyar nyelvre az általunk fejlesztett jegyek hozzáadásával további eredményjavulást értem el.

Végül végeztem hibaanalízist is angol-magyar nyelvre. Megvizsgáltam, hogy milyen esetekben rontott a kompozit rendszerem. Sok esetben fordult elő olyan hiba, hogy a három gépi fordítás teljesen megegyezett, de az automatikus kiértékelő rendszerektől különböző értékeket kaptak. Mivel a három minőségbecslő modellt külön tanítottam

DOI:10.15774/PPKE.ITK.2019.003

		en-hu	en-hu+
Korreláció ↑	PBMT	0,6667	**0,6884**
	HBMT	0,5926	**0,6199**
	NMT	0,5926	**0,6199**
MAE ↓	PBMT	0,1809	**0,1730**
	HBMT	0,1953	**0,1888**
	NMT	0,1953	**0,1888**
RMSE ↓	PB	0,2266	**0,2196**
	HB	0,2402	**0,2341**
	NMT	0,2402	**0,2341**

6.3. táblázat Angol-magyar modellek teljesítménye az általam fejlesztett jegyek hozzáadásával

be, ezért azok is különböző értékeket eredményeztek. A 6.4. táblázatban erre látható egy példa. Az „Oil ring" fordításai teljesen megegyeznek, a minőségbecslés alapján az NMT fordítása a legjobb, az automatikus metrikák szerint viszont a PBSMT vagy a HBSMT. A „Corrosion of terminal" példa esetében pedig minimális az eltérés a fordítások között, de jelentésben szinte megegyeznek. A minőségbecslés alapján az NMT a legjobb, viszont az automatikus kiértékelés szerint nem az. A „Bulb type brake light" példa esetén a rendszerem a HBSMT fordítását választotta a legjobbnak, míg az automatikus kiértékelés alapján az NMT fordítása a legjobb. Megvizsgálva ezeket a példákat, az látható, hogy a fordítások jelentésben vagy megegyeznek, vagy nagyon hasonlóak. Habár a rendszerem a legjobbnak nem azt választotta, amit az automatikus kiértékelő módszer, mégis jelentésben és olvashatóságban jók azok a fordítások. Ez a típusú hiba több, mint az esetek felét teszi ki. Ez alapján feltételezhető, hogy a 6.2. táblázatban látható CoMT értékek jóval magasabbak a mostani értékeknél. Ez a vizsgálat arra mutat rá, hogy az automatikus referenciafordítással történő kiértékelő módszerek nem mindig tükrözik hűen a fordítás minőségét.

6.5. Továbblépési lehetőségek

Kutatásomban a vállalati környezethez alkalmazkodtam, ezért automatikus kiértékelési mértékeket alkalmaztam a minőségbecslő modellek tanításához. Az automatikus módszerek egyik hátránya, hogy alacsonyan korrelálnak az emberi kiértékelésekkel. Amíg viszont

DOI:10.15774/PPKE.ITK.2019.003

	oTER ↓	QE ↓	
Forrás	-	-	Oil ring
NMT	0,129	**0,117**	Olajlehúzó gyűrű
PBSMT	0	0,173	Olajlehúzó gyűrű
HBSMT	0	0,135	Olajlehúzó gyűrű
Forrás	-	-	Corrosion of terminal
NMT	0,385	**0,148**	Az érintkező korróziója
PBSMT	0,236	0,285	Érintkezők korróziója
HBSMT	0,236	0,265	Érintkezők korróziója
Forrás	-	-	Bulb type brake light
NMT	0,211	0,165	Izzóval szerelt féklámpa
PBSMT	0,688	0,167	Féklámpa izzó típus
HBSMT	0,381	**0,112**	Izzós féklámpa

6.4. táblázat Hibák elemzése

nincsen jobb, megbízható módszer, addig a vállalatok az automatikus kiértékelési módszereket fogják használni. Ha lenne rá lehetőségem, akkor mindenképpen kísérleteznék, vállalati környezetben, emberi kiértékelésre betanított minőségbecslő modellekkel.

Egy másik irány a neurális alapú minőségbecslő modell alkalmazása lenne, amely egyelőre egy megbízható minőségű angol-magyar neurális minőségbecslő rendszer nélkül nem jöhet létre.

Végül, de nem utolsó sorban, érdekes lehet egy olyan kísérlet, ahol egy minőségbecslő modellt tanítok be a három gépi fordítórendszernek. Feltételezésem szerint egy általánosabb minőségbecslő modell gyengébben teljesít, mint a külön-külön betanított és optimalizált minőségbecslő modellek.

6.6. Összegzés

Létrehoztam egy kompozit gépi fordítórendszert, amely a minőségbecslés módszerével különböző gépi fordítórendszerek kimeneteit kombinálva ér el rendszerszinten jobb eredményt, mint az általa felhasznált gépi fordítórendszerek önmagukban.

Kutatásomban kombináltam egy kifejezésalapú statisztikai, egy hierarchikus statisztikai és egy neurális gépi fordítórendszer kimeneteit. A kombináláshoz mondatszintű minőségbecslés módszerét alkalmaztam. Mindegyik gépi fordítórendszerhez külön-külön betanítottam egy minőségbecslő modellt. A tanításhoz csak „black-box” jegyeket hasz-

DOI:10.15774/PPKE.ITK.2019.003

náltam, valamint a vállalati környezethez alkalmazkodva, automatikus mértékekre tanítottam be a modelleket. Háromféle automatikus mértéket használtam: BLEU, ortho-BLEU és orthoTER.

Kutatásomat négy különböző nyelvpárra végeztem el: angol-magyar, angol-német, angol-olasz és angol-japán.

Az eredmények alapján rendszerszinten a kompozit gépi fordítórendszerem minden esetben jobb minőséget eredményezett, mint a PBSMT, a HBSMT és a NMT rendszerek önmagukban. Angol-magyar nyelvpár esetében nyelvfüggő jegyekkel tovább tudtam növelni a rendszer minőségét.

Kapcsolódó tézis

4. tézis: **Létrehoztam egy kompozit gépi fordítórendszert, amely a minőségbecslés módszerével, különböző gépi fordítórendszerek kimenetét kombinálva ért el rendszerszinten jobb eredményt, mint az általa felhasznált gépi fordítórendszerek önmagukban.**

A tézishez kapcsolódó publikációk: [5] [6] [12].

DOI:10.15774/PPKE.ITK.2019.003

7. fejezet

A πRate rendszer

7.1. Előzmények

Kutatásomban a minőségbecslés módszerét alkalmaztam egynyelvű szövegek minőségének becslésére és hibáinak detektálására.

Az internet elterjedésével manapság egyre nagyobb számban állnak rendelkezésünkre korpuszok és nyersanyagok nyelvtechnológiai vagy nyelvészeti kutatásokhoz. Azonban sok esetben az elérhető szövegek nem a kutatók által ismert vagy megszokott formátumokban vannak. Ez rendkívül megnehezíti a feladatukat. Ezért, hasonlóan a gépi fordítás minőségbecslésének esetéhez, mind az előfeldolgozás, mind a feldolgozás fázisában rendkívül fontosak lehetnek a minőségi mutatók, amelyek információt biztosítanak a szövegek minőségéről. Ha egy minőségbecslő rendszer, a minőség mértéke mellett, a hiba jellegét is meg tudja becsülni, az igen fontos információt jelenthet a kutatók számára, hiszen ez a szöveg feldolgozásának módszerét is meghatározhatja. Továbbá egy jól működő, egynyelvű minőségbecslő rendszer segítséget nyújthat egy nyelvi elemző rendszer számára is. Egy megbízható minőségi mutató segíthet az automatikus természetesnyelv-feldolgozó szoftvereknek a döntéstámogatásban és feldolgozási vagy normalizációs módszer kiválasztásában. Hasznos segédeszköz lehet a korpuszokat vizsgáló nyelvészkutatók számára is.

Az egynyelvű szövegek minőségének meghatározása komoly kihívás. A gépi fordítástól eltérően, az emberek más jellegű hibákat vétenek.

DOI:10.15774/PPKE.ITK.2019.003

A hagyományos minőségbecslő rendszerek első sorban gépi fordítás minőségének becslésére fókuszálnak. A QuEst++ [74] rendszer szószintű elemző része tartalmaz egynyelvű kiértékeléseket, többek között nyelvimodell-jegyeket, szintaktikai jegyeket, célnyelvi kontextus jegyeket stb. De ezek csupán egy apró részét képezik a teljes rendszernek, amely nem kifejezetten egynyelvű szövegek minőségbecslésének céljával készült. Ezért első feladatom az volt, hogy a gépi fordítás minőségbecslő rendszerét egynyelvű szövegek minőségének becslésére módosítsam. Továbbá, hogy úgy továbbfejlesszem a rendszert, hogy egy nyelvi elemző előfeldolgozó részévé válhasson. Végül kísérleteket végeztem olyan modellek létrehozásával, amelyek alkalmasak az egynyelvű szövegek minőségének meghatározására.

Kutatásom célja az volt, hogy megvizsgáljam az interneten elérhető emberek által készített szövegek hibáit, valamint – a minőségbecslés módszerével – létrehozzak egy automatikus hibadetektáló programot. A kutatást Dömötör Andrea nyelvésszel közösen végeztem el.

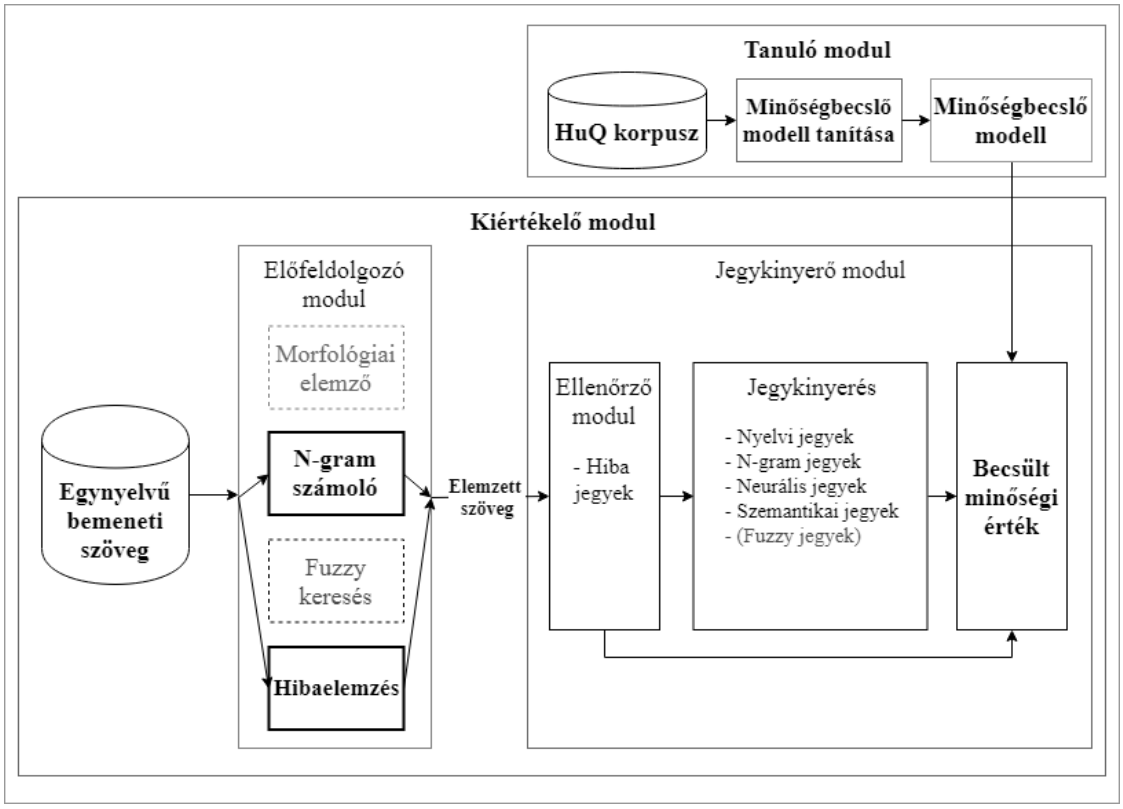
7.2. Egynyelvű minőségbecslő rendszer

Létrehoztam egy egynyelvű minőségbecslő rendszert, a πRate rendszert. A módszer a gépi fordításhoz használt minőségbecslő rendszer architektúráját (lásd 7.1. ábra) veszi alapul.

A rendszernek két fő modulja van: a tanuló modul és a kiértékelő modul. A tanuló modul feladata, hogy betanítsa a minőségbecslő modellt. Ez a modul működésében megegyezik a hagyományos gépi fordításnál használt minőségbecslő modell tanuló moduljával. A tanításhoz a Dömötör Andrea által készített egynyelvű korpuszt használtam. A gépi tanuláshoz különböző nyelvi és statisztikai jegyeket használtam, mint például: nyelvi jegyek, nyelvmodell jegyek, hibajegyek stb. A korpuszból a jegyek segítségével kinyertem a minőségi mutatókat, majd a mutatók segítségével betanítottam a minőségbecslő modellt az emberek által kiértékelt minőségi mutatókra.

A másik fő modul a kiértékelő modul. Ez először beolvassa a bemeneti szöveget, majd az előfeldolgozó fázisban elemzi azt. Az így elemzett szöveget adja tovább a jegykinyerő modulnak, amely a különböző jegyek és a betanított minőségbecslő modell segítségével előállítja a minőségi mutatókat. A kiértékelő modulban használt jegyek megegyeznek

DOI:10.15774/PPKE.ITK.2019.003



7.1. ábra The πRate rendszer architektúrája

a tanító modulban használt jegyekkel. A bemenet lehet nyers, vagy elemzett szöveg. A rendszer egyik újító tulajdonsága, hogy feladatorientált architektúrával rendelkezik. Ezért, ha a bemenet már elemzésre került, akkor a morfológiai elemzés művelete kihagyható az előfeldolgozó fázisban. Így az erőforrás, ezáltal a teljesítmény optimalizálható. A kiértékelő modulnak három fő része van: *előfeldolgozó modul*, *ellenőrző modul* és *jegykinyerő modul*.

Amint a szöveg beérkezik a rendszerbe, az előfeldolgozó modul morfológiailag elemzi a szöveget (ha az még nem került elemzésre), kiszámolja az n-gram valószínűségeket stb. Majd az elemzett szöveget továbbküldi a jegykinyerő modul számára. Itt először az ellenőrző modul ellenőrzi le azt, a hibajegyek segítségével. Ha a szöveg hibamértéke meghalad egy megadott küszöbértéket, akkor az ellenőrző modul bizonyos feltételek esetén megszakíthatja a folyamatot, vagy szűrheti, „cenzúrázhatja” a szöveget. Máskülönben továbbengedi a többi jegy számára, és a minőségbecslő modell a saját hibaértékeit használja fel minőségi mutatóként. A jegyek kinyerése után a minőségbecslő modell a

DOI:10.15774/PPKE.ITK.2019.003

minőségi mutatók alapján kiszámolja a becsült értékeket (pl.: aktuális mondat minősége, eddig beolvasott összes szöveg globális minősége stb.). A rendszer képes inkrementálisan növekvő bemenetet kezelni.

A πRate rendszer implementálásához JAVA EE-t (Enterprise Edition) használtam.

7.3. Felhasznált korpusz és jegyek

A minőségbecslő modell felépítéséhez egynyelvű jegyekre volt szükség, amelyek segítségével a rendszer kinyeri a minőségi mutatókat. A tanítás során a jegyek egy egynyelvű korpuszból nyerték ki a szükséges értékeket. Majd gépi tanulással emberek által kiértékelt minőségi mutatókra tanítottam be a minőségbecslő modellt (lásd 7.1. ábra).

A kutatásomhoz Dömötör Andrea nyújtott segítséget, ő készítette a korpuszt. Én készítettem az egynyelvű jegyeket a modell tanításához, valamint a minőségbecslő rendszert. Az alábbiakban bemutatom az általam használt korpuszt és jegyeket.

7.3.1. Egynyelvű korpusz

A korpuszt teljes egészében Dömötör Andrea készítette.

Az egynyelvű, emberek által létrehozott szövegek hibái más jellegűek, mint azok, amelyeket egy gépi fordító követ el, ezért az egynyelvű szövegek minőségbecslése másfajta tanítóanyagot és jegykészletet igényel. A tanító- és tesztkorpusz az MNSz2 korpuszból [105] készült, amely a beszélt nyelvi és a személyes alkorpuszokból véletlenszerűen lekért adatokból áll. Azért esett a választás ezekre a szövegtípusokra, mert feltételezhetően ezekben a leggyakoribb a sztenderdtől való eltérés. A tanítókorpusz annotálása nyelvészeti alapokon, manuálisan történt. Kétféle annotációból áll: Likert-skála (1-5) és osztályozási címkék. Összesen 1024 darab annotált mondat született.

A Likert-pontszámok esetén nem a szubjektív emberi értékelés volt a szempont, hanem az, hogy a mondat elemzése várhatóan mennyire okozhat nehézséget (a normától való eltérésből adódóan) egy szabály alapú gépi eszköz számára. A pontszámok kiszámolása a helyesen elemezhető összetevők és összetevős szerkezetek arányából történt. Elemzendő összetevők alatt az NP-k és névutós szerkezetek, az igék, igekötők és vonza-

DOI:10.15774/PPKE.ITK.2019.003

tok kapcsolata, illetve a tagmondatok, valamint a teljes mondat értendő. Az (1a)-ban és (1b)-ben látható példákban [] jelöli a felismerhető, és []! a nem felismerhető összetevőket.

(1) a. *[Emberünk ugyanis [állateledel és kisállat kereskedést [tart fenn]]! .]!*

b. *[Ugyanis [törvénysértést [követett el]] [az erkölcsrendész ismerosöm szerint]!]!*

Az osztályozás során a hibátlan mondatok kaptak 5 pontot, a 20%-nál kevesebb felismerhetetlen összetevőt tartalmazók 4-et, a 20 és 39% közötti hibaarányúak 3-at, a 40 és 59% közöttiek 2-t. A legrosszabb 1-es pontszámot azok a mondatok kapták, ahol az összetevők legalább 60%-a minősült elemezhetetlennek. Ez az annotálási rendszer bonyolultnak és indokolatlanul időigényesnek tűnhet, de például (1a)-ban látható, hogy az *állateledel- és kisállatkereskedés* rossz helyesírásán az emberi értelmezés ugyan könnyen túllendül, egy számítógépes elemző számára azonban gyakorlatilag értelmezhetetlen (vagy félreérthető) így a mondat. Az értékelés tehát azért nem pusztán emberi ítélettel történt, mert a pontszámokkal elsősorban gépi eszközöket szeretnénk informálni, így a pontozásnál ki kellett iktatni az ember „természetes normalizáló képességét".

Az így kapott érték információt adhat az elemző rendszernek az input megbízhatóságáról. Ugyanakkor a hiba típusa még sokkal informatívabb lehet egy gépi eszköz számára: ha a minőségbecslő megbízhatóan detektálja a hiba jellegét, az eszköz alkalmazni tudja a megfelelő normalizáló modult (például: helyesírás-ellenőrző, ékezet-visszaállító).

Az osztályozási modell hét osztálycímkét tartalmaz. A létrehozott osztályok a következők:

1. Központozás hibái (hiánya), nagybetűk elhagyása (KH);
2. Elírások, helyesírási és nyelvi hibák (HH);
3. Idegen nyelvű, idegen szavakat tartalmazó szövegek (INY);
4. Ékezetek hiánya (ÉH);
5. Nehezen elemezhető beszélt nyelvi vagy informális szövegek (ismétlések, elakadások, szleng, rövidítések, emotikonok stb.) (BNY);

DOI:10.15774/PPKE.ITK.2019.003

6. Szegmentálási hiba-osztály (SZH);

7. Hibátlan mondat (Helyes).

Az első öt hibatípus a beszélt nyelvi és az informális internetes szövegekre jellemző. A szegmentálási hibaosztály alatt azok az esetek értendők, amikor a korpuszból kapott adat valójában nem volt mondat, vagy nem egy mondat volt.

Ha egy mondat több hibatípusba is besorolható, az egycímkés tanításhoz, egy fő hiba került kiválasztásra: az, amelyik a szöveg nagyobb részét lefedi. Például (2a)-ban elírás is szerepel (*gyeppet*), mégis jobban jellemzi a szöveget az ékezetek hiánya. (2b)-ben pedig, bár egy mondatközi írásjel is hiányzik, de jelentősebb a helyesírási hibák előfordulása, hiszen a mondat nyolc szavából három is hibás.

(2) a. *De gyeppet sem siettek el a dolgot, es mindharman a jarda kozepen tanyaztak.*
Ékezetek hiánya

b. *Valamelyik ujságban olvastam hogy állitolag fel akarják újitani.*
Helyesírási hibák

A többcímkés osztályozási modellhez minden mondat 3 címkét kapott, ennél több hibaosztályba egyik adat sem volt besorolható. Ahol háromnál kevesebb hibatípus fordult elő, ott a hiányzó helyekre a „helyes” címkét kapták a mondatok.

7.3.2. Egynyelvű jegyek

A minőségbecslő modellem 36 különböző jegyet használ, amelyeket jellegük alapján az alábbi kategóriákba soroltam:

- nyelvi jegyek:
  - főnevek, igék, igekötők, melléknevek, határozószók, kötőszók, névmások, névelők, indulatszók aránya a mondatban;
  - főnevek és igék aránya a mondatban;
  - főnevek és melléknevek aránya a mondatban;
  - igék és igekötők aránya a mondatban;
  - főnevek és névelők aránya a mondatban;

DOI:10.15774/PPKE.ITK.2019.003

  - tokenek száma;
  - átlagos szóhossz a mondatban;

- n-gram jegyek:
  - a mondat n-gram valószínűsége;
  - a mondat n-gram perplexitása;
  - a mondat szótöveinek, szófaji címkéinek n-gram valószínűsége;
  - a mondat szótöveinek, szófaji címkéinek n-gram perplexitása;

- neurális nyelvmodell jegyek:
  - 1-gram, 2-gram és 3-gram perplexitás;

- hibajegyek:
  - ismeretlen szavak aránya a mondatban;
  - ékezetes szavak aránya a mondatban;
  - írásjelek aránya a mondatban.

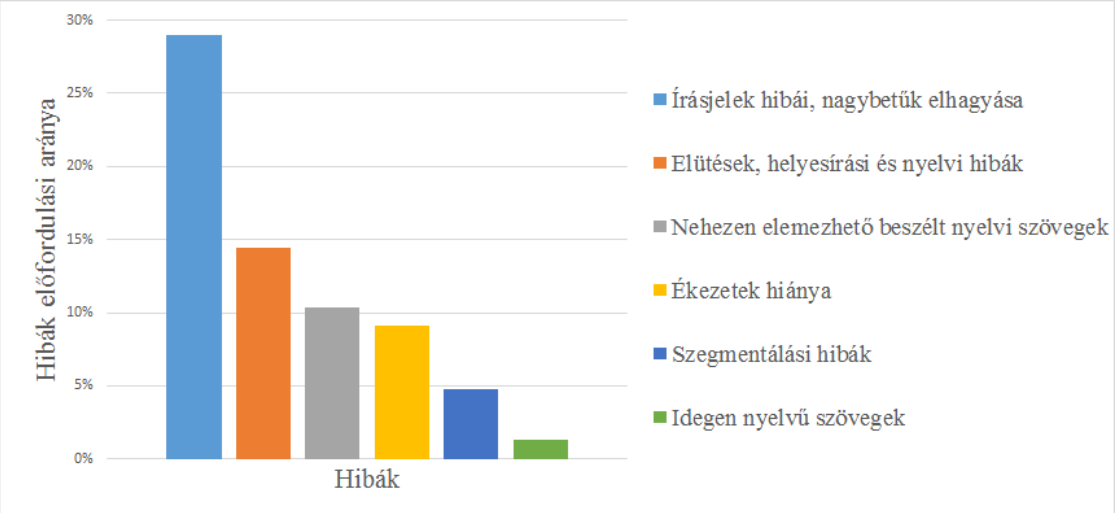
Az n-gram modell felépítéséhez (az n-gram jegyekhez) szintén az MNSz2 egy részkorpuszát használtam, amely 98500 lemmatizált és elemzett mondatot tartalmazott.

A neurális nyelvmodell tanításához a Pázmány Korpuszból [106] vettem 1 millió mondatot. A nyelvmodell felépítéséhez egy GRU-alapú (Gated Recurrent Unit) RNN architektúrát használtam, amely 6 epochig tanult. Továbbá a modellemhez szóbeágyazást is alkalmaztam, amelyhez a [84] által készített magyar nyelvű szóbeágyazási modellt használtam fel.

7.4. Mérések és eredmények

Kutatásomban először a hibák előfordulási arányát vizsgáltam meg (lásd 7.2. ábra). A legtöbbször előforduló hiba az „írásjelek hibái, nagybetűk elhagyása” hibatípus volt. Ez azt jelenti, hogy ha kezelnénk ezt a hibatípust, a hibák ∼30%-át megoldanánk. Viszont ahhoz, hogy kezelni tudjuk a hibát, szükség van egy szoftveres megoldásra, ami meg tudja állapítani a hiba típusát.

DOI:10.15774/PPKE.ITK.2019.003



7.2. ábra A hibák előfordulási aránya

A kutatásom során a gépi tanulás módszerével kétféle modellt készítettem: osztályozási és regressziós. A Likert értékek segítségével regressziós modelleket építettem, a hibaosztályok segítéségével pedig osztályozási modelleket készítettem.

Az osztályozás esetében végeztem egycímkés (a fő hibaosztállyal) és többcímkés osztályozást is. A regresszióhoz és az egycímkés osztályozáshoz a Weka [75] szoftvert, míg a többcímkés osztályozáshoz a MEKA [107] rendszert alkalmaztam.

Többféle tanuló algoritmust is kipróbáltam, melyek közül az egycímkés osztályozáshoz a szupport vektor gép, a regresszióhoz a szupport vektor regresszió, a többcímkés osztályozáshoz pedig a véletlen erdő (random forest) alapú „Classifier Chains” [108] módszer érte el a legjobb eredményt. Az eredmények fejezetben csak az ezekkel a módszerekkel betanított modellek eredményeit mutatom be. Az annotált tanítóanyag segítségével az alábbi három minőségbecslő modellt építettem:

- LS modell: regressziós minőségbecslő modell, a Likert értékeket felhasználva. A Likert értékek, 1-től 5-ig terjedő egész számok.
- CS modell: egycímkés osztályozási minőségbecslő modell, a fő hibaosztályokat felhasználva. Összesen 6 hibaosztály, és a helyes mondat osztályozási címkéje.

DOI:10.15774/PPKE.ITK.2019.003

- CCS modell: többcímkés minőségbecslő modell, az osztályzási értékeket felhasználva. Minden mondathoz 3 db osztályt rendeltem, az első osztály a főcímke, ami vagy a fő hibaosztály, vagy a helyes mondat címkéje. A második és a harmadik osztály a mellékhibá(ka)t tartalmazza. Ha nincs ilyen, a helyes mondat címkéjét viselik.

Továbbá végeztem optimalizációt is. A gépi fordítás kiértékelésének optimalizációja alapján [76], ha kiveszem a kevésbé releváns jegyeket a rendszerből, kevesebb jeggyel magasabb minőséget lehet elérni. Az angol-magyar minőségbecslés kutatásából kiindulva, a „forward selection" módszerével (lásd 5.4. fejezet), az alábbi optimalizált modelleket hoztam létre:

- OptLS modell: optimalizált LS modell.
- OptCS modell: optimalizált CS modell.

7.4.1. Fuzzy jegyek kísérlete

A Hanna Bechara és társainak szemantikai hasonlóság kutatása [109] alapján végeztem egy olyan kísérletet is, ahol a HuQ korpusz egyharmadát referenciakorpuszként használtam fel. A referenciamondatok közül fuzzy kereséssel megkerestem a bemeneti szöveghez legjobban hasonlító mondatot, majd a megtalált referenciamondathoz tartozó minőségi értékeket (Likert és osztályzási értékei) a minőségbecslő modellemben felhasználtam minőségi mutatóként (jegyként). A fuzzy kereséshez kipróbáltam a Levenstein távolságot, a TER (Translation Error Rate) mértéket, a BLEU mértéket, a NIST mértéket, a szemantikai hasonlóságot mérő LSI módszert és a szóbeágyazási modelleket. A Levenstein, a TER, a BLEU és a NIST módszerek esetében általában több találat is volt. Ezekben az esetekben az LSA és a szóbeágyazási modellek segítségével szűkítettem a találatokat.

Kísérletemben a HuQ korpuszt felosztottam: 500 mondatot fuzzy referenciához, 1000 mondatot minőségbecsléshez soroltam. A reprezentativitás érdekében az 500 mondatos fuzzy referenciakorpuszt kézzel állítottam össze, amely közel egyenlő arányban tartalmazott „BAD", „MEDIUM" és „GOOD" osztályzatú mondatokat (167 „BAD" mondat, 166 „MEDIUM" mondat, 167 „GOOD" mondat).

DOI:10.15774/PPKE.ITK.2019.003

Az 1000 mondatot, amelyeket a minőségbecslő modellhez soroltam, további 90-10% arányba osztottam fel tanító és tesztelő halmazra. A teszteléshez 10-szeres keresztvalidálást használtam.

Ebben a mérésben több Fuzzy jegy is releváns volt az eredményre nézve (lásd D. függelék). Mivel a HuQ korpuszban gép által elkövetett hibák szerepelnek, és nem emberi hibák, ezért az emberi hibák detektálásában nem használtam fel ezeket a jegyeket.

Ebben a kutatásban végzett eredmények a D. függelékben találhatóak.

7.4.2. Eredmények

Először elvégeztem a kísérletet a megfelelő gépi tanuló algoritmus kiválasztására. A 7.1. és a 7.2. táblázatban látható, hogy regresszióra az SVR, míg osztályozásra a véletlen erdő módszerek érték el a legjobb eredményeket. Ezért kutatásom további részében ezt a két módszert használtam.

	Korreláció ↑	MAE ↓	RMSE ↓
Lineáris regresszió	0,7240	0,8037	1,0523
Gaussi eljárás	0,7301	0,8290	1,0331
SVR	0,7712	0,7121	1,0047

7.1. táblázat Tesztelt algoritmusok regresszióra

	CCI ↑	MAE ↓	RMSE ↓
Döntési fa	57,07%	0,1288	0,3292
SVM	62,83%	0,2141	0,3173
Véletlen erdő	64,48%	0,2140	0,3171

7.2. táblázat Tesztelt algoritmusok osztályozásra

Továbbá megvizsgáltam a rendszer teljesítményét. A kiértékeléshez a MAE, az RMSE, a Pearson-féle korreláció (Korreláció), a helyesen osztályozott egyed (CCI). A teszteléshez minden esetben 10-szeres keresztvalidálást használtam. A 7.3. és a 7.4. táblázatban látható, hogy a 36-os jegykészlet ∼77,1%-os korrelációt és ∼64,48% helyesen osztályozott egyedet ért el.

Az optimalizálás utáni eredményeket a 7.3. és a 7.4. táblázat második sorai mutatják be.

DOI:10.15774/PPKE.ITK.2019.003

	Korreláció ↑	MAE ↓	RMSE ↓
LS modell - 36 jegy	0,7712	0,7121	1,0047
OptLS készlet - 15 jegy	0,7777	0,7226	0,9625

7.3. táblázat Az LS modell és az OptLS jegykészlet értékelése

	CCI ↑	MAE ↓	RMSE ↓
CS modell - 36 jegy	64,48%	0,214	0,3171
OptCS készlet - 28 jegy	65,17%	0,2137	0,3167

7.4. táblázat A CS modell és az OptCS jegykészlet értékelése

- Az OptLS jegykészlet, amelyik 15 jegyet használ, lényegesen kevesebb számítással, közel azonos korrelációt ért el, mint a teljes jegykészlet.
- Az OptCS jegykészlet, amelyik 28 jegyet használ, kevesebb munkával, közel azonos eredményt ért el, mint a teljes jegykészlet.

Amint látható, a Likert-skála modell megfelelő korrelációval működik, ám az egycímkés osztályozási modell kevesebb eredményességet mutat. Ennek magyarázata részben az annotálási módszer lehet. Amint a 7.3.1 alfejezetben írtam, az annotáció szerint, egy mondat csak egy hibaosztályba tartozhat, holott a valóságban többféle hibát is tartalmazhat. Azaz, az osztályozó feladata valójában az, hogy meghatározza az elsődleges hibatípust. Ez viszont igencsak nehéz lehet, ha a mondatban egyéb, kevésbé releváns hibák is megtalálhatók.

A sikertelenség okainak pontosabb feltérképezésére a tévesztési mátrix (lásd 7.5. táblázat) adhat választ. Az oszlopok jelölik a gép által predikált értékeket, a sorok pedig azt, hogy mit kellett volna. Ebből látható, hogy a rendszer jól boldogul az ékezet nélküli és a hibátlan szövegek besorolásával, a többi osztálynál viszont gyengébb eredményeket mutat. Különösen figyelemre méltó, hogy az idegen nyelvű (INY) és a beszélt nyelvű (BNY) szövegekre egyszer sem becsült a program. Úgy látszik, a rendszer által használt jegyekkel, ezek jellemezhetők a legkevésbé.

Következő lépésként megvizsgáltam a hibatípusok közötti összefüggéseket.

DOI:10.15774/PPKE.ITK.2019.003

	ÉH	HH	INY	BNY	KH	SZH	Helyes	F-mérték
Ékezet (ÉH)	107	0	0	0	1	0	7	0,915
Helyesírás (HH)	2	30	0	0	40	1	72	0,267
Idegen nyelvű (INY)	1	0	0	0	1	3	3	0,0
Beszélt nyelvi (BNY)	0	6	0	0	28	0	19	0,0
Központozás (KH)	3	23	0	0	127	1	83	0,547
Szegmentálás (SZH)	5	10	0	0	14	12	21	0,304
Helyes	1	3	0	0	16	0	385	0,774

7.5. táblázat Tévesztési mátrix

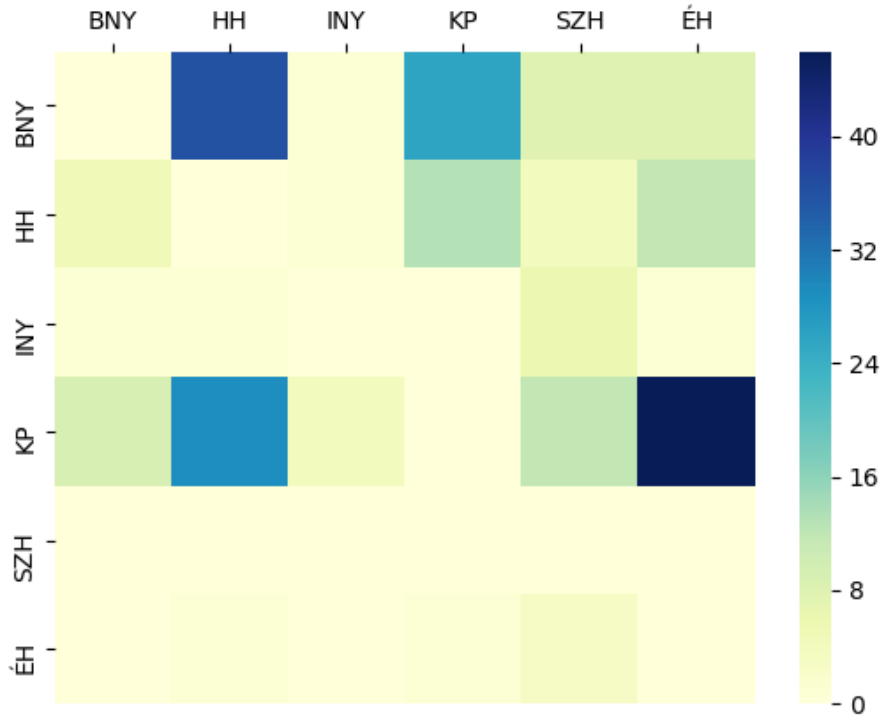
	Átlagos pontszám
Ékezetek hiánya	1,16
Idegen nyelvű szövegek	1,63
Szegmentálási hibák	1,74
Elütések, helyesírási és nyelvi hibák	2,69
Írásjelek hibái (hiánya), nagybetűk elhagyása	3,20
Nehezen elemezhető beszélt nyelvi szövegek	3,28

7.6. táblázat A hibaosztályok átlagos Likert-pontszáma

A 7.6. táblázat az egyes hibaosztályok átlagos Likert pontszámát mutatja. Eszerint, a normalizálást tekintve, az ékezet-visszaállítás, a nyelvfelismerés, valamint a megfelelő mondatokra és tagmondatokra bontás (praktikusan írásjel-visszaállítás) jelentheti a legnagyobb segítséget egy gépi feldolgozó eszköz számára.

Továbbá megvizsgáltam a főhibák és a mellékhibák együttes előfordulását is (lásd 7.3. ábra). Az ábrán a sorok jelentik a főhibát, az oszolopok pedig azt jelzik, hogy az adott mellékhiba melyik főhibákkal fordul elő. Az ábrából látszik, hogy az írásjelekkel kapcsolatos főhibával mindegyik hiba előfordul mellékhibaként . Ez azt jelenti, hogy amikor nem figyelünk az írásjelekre, akkor hajlamosak vagyunk további hibákat is véteni. Ez a jelenség a nehezen elemezhető beszélt nyelvi hibáknál és a helyesírási hibáknál a leglátványosabb. Továbbá az is látható, hogy a beszélt nyelvi hibák erősen összefüggnek a helyesírási hibákkal. Érdekes továbbá, hogy az ékezetek hiánya is nagyrészt következményként jelentkezik, vagyis az esetek többségében mellékhibaként van jelen, főhibaként nagyon ritkán. Ez a jelenség igaz a szegmentálási hibákra és az idegen nyelvű szövegekre is.

DOI:10.15774/PPKE.ITK.2019.003



7.3. ábra A főhibák és a mellékhibák együttes előfordulása

Megvizsgáltam a hibák együttes előfordulásának viselkedését egy másik szemszögből is a főkomponens analízis módszerével. Ehhez készítettem egy mátrixot, amely szerkezetében megegyezik az együttes előfordulási mátrixszal, annyi különbséggel, hogy a cellákban nem az előfordulás mennyisége szerepel, hanem 0 vagy 1 érték, amely azt jelöli, hogy előfordulnak-e együtt, vagy nem. A kapott eredményeket a 7.7. táblázat tartalmazza. Ebből az derült ki, hogy a vizsgált informális műfajokban a kevésbé gondos szövegalkotás több jelenségben is megnyilvánulhat egyszerre. Így például számítani lehet arra, hogy az ékezeteket nem tartalmazó szöveg – nagy valószínűséggel – nyelvi szempontból sem fog megfelelni a sztenderdnek (l. 1. faktor). A szegmentálási hibák, az elemzés szerint, gyakran az idegen nyelvű szövegekkel járnak együtt (l. 2. faktor). Az írásjelek elhagyása viszont, úgy tűnik, a többi szempontból kifogástalan szövegekre is nagy arányban jellemző (3. faktor). Mivel az egycímkés osztályozási modell tanuló korpusza jelenleg csak egy – elsődlegesnek tekintett – hibatípust rendel a mondatokhoz, fontos lehet ezeknek az összefüggéseknek az ismerete.

DOI:10.15774/PPKE.ITK.2019.003

1. faktor	beszélt nyelvi szöveg	nyelvi hibák	ékezetek hiánya
2. faktor	szegmentálási hibák	idegen nyelvű szövegek	
3. faktor	központozási hibák		

7.7. táblázat A hibatípusok összefüggései főkomponens analízissel

7.4.3. Többcímkés osztályozási modell

A többcímkés osztályozási modell (lásd 7.8. táblázat) a fő hibaosztály detektálásában hasonló eredményességet mutat, mint az egycímkés modell. A második és harmadik címke pontosságának javulása annak is köszönhető, hogy ezek egyre nagyobb arányban tartoztak a hibátlan osztályba. Mindemellett az 56,6%-os pontos egyezés, a feladat komplexitását és a tanuló adatok kis számát tekintve, jó eredménynek tekinthető.

	Fő hibaosztály	2. hibaosztály	3. hibaosztály
Pontosság címkénként ↑	0,652	0,835	0,964
Pontos egyezés ↑		0,566	
Hamming veszteség ↓		0,183	

7.8. táblázat A többcímkés osztályozási modell eredményei

7.4.4. Az optimalizált jegykészlet

A 7.9. és 7.10. táblázatokban látható az optimalizált jegykészletek első 10 eleme, a jegyek relevanciája szerint rendezve. A teljes optimalizált jegykészletek a C.2. függelék C.2. és C.3. táblázataiban találhatóak.

A Likert-skála szerinti minőségbecslés szempontjából leginkább releváns nyelvi és hibajegyek az ékezetekkel, a tokenszámmal és az írásjelekkel függnek össze. Ez az eredmény nyelvészeti szempontból nézve nem meglepő. Az informális írásbeli kommunikációban (azaz legjellemzőbben az internetes szövegekben) az ékezetek és írásjelek elhagyása a legtipikusabb sztenderdtől való eltérés. Utóbbi a gépi feldolgozásban szegmentálási problémákat okozhat, ezzel magyarázható a sokszor extrém magas tokenszám. Ha tehát egy mondat – vagy amit az elemző egy mondatnak hisz – nagyon hosszú, a minőségbecslő rendszer arra a következtetésre juthat, hogy egy szerkesztetlen szövegről van szó, ami

DOI:10.15774/PPKE.ITK.2019.003

Jegy
Ékezetes karakterek száma a mondatban.
Ékezetes szavak száma / tokenek száma a mondatban.
A mondat szavainak perplexitása (ismeretlen szavakkal együtt).
A mondat elemzési címkéinek perplexitása (ismeretlen szavakkal együtt).
A mondat 1-gram perplexitása (neurális nyelvmodell).
A mondat elemzési címkéinek perplexitása (ismeretlen szavak nélkül).
Ismeretlen szavak aránya a mondatban.
A mondat szótöveinek perplexitása (ismeretlen szavak nélkül).
A mondat szófajcímkéinek perplexitása (ismeretlen szavakkal együtt).
A mondat szótöveinek perplexitása (ismeretlen szavakkal együtt).

7.9. táblázat A Likert-modellhez optimalizált jegykészlet első 10 eleme

Jegy
Ékezetes szavak száma / tokenek száma a mondatban.
Írásjelek aránya a mondatban.
Ékezetes karakterek száma a mondatban.
Névmások aránya a mondatban.
A mondat szótöveinek n-gram valószínűsége.
Igék aránya a mondatban.
Főnevek száma / névelők száma.
Főnevek száma / igék száma.
Tokenek száma a mondatban.
A mondat elemzési címkéinek n-gram valószínűsége.

7.10. táblázat Az egycímkés osztályozási modellhez optimalizált jegykészlet első 10 eleme

gyakran együtt jár az alacsony minőséggel. Az optimalizált jegykészletben előfordul még az ismeretlen szavak és az indulatszók aránya is. Ezen címkék magas száma szintén az informális szövegek sajátossága.

Az egycímkés osztályozási modell optimalizált jegyei között már több olyan jegy is megjelenik, amely valódi nyelvi hibára utalhat. Ilyenek lehetnek például a főnevek és névelők, az igék és igekötők aránya, vagy a névmások száma. Ezek szükségesek ahhoz, hogy a modell a nyelvtani helyességre (helytelenségre) vonatkozó hibaosztályokat is detektálni tudja. Az ilyen típusú jegyek azonban kevésbé tűnnek relevánsnak a Likert-modell esetén. Ez azt mutatja, hogy az egynyelvű korpuszokból származó szövegek minőségi problémái nagyrészt nem nyelvi természetűek, a szó szoros értelmében.

DOI:10.15774/PPKE.ITK.2019.003

7.5. Továbblépési lehetőségek

A rendszer értékeléséből az derült ki, hogy a hibatípusok detektálása még továbbfejlesztésre szorul. Egyelőre csak a hibátlan és az ékezet nélküli szövegek elkülönítésében működik megbízhatóan. További releváns jegyek kutatásával tovább növelhető a rendszer hibafelismerő képességének pontossága.

A minőségbecslés módszerét kiterjesztettem egynyelvű szövegek stílusának osztályozására. Azonban ebben a feladatban, a gépi tanulás módszere – szemben a neurális módszerekkel – rendkívül alacsonyan teljesített [2]. Ezért ez irányban a minőségbecslés módszerével zsákutcába futottam. Ehelyett a neurális módszerekre helyeztem a kutatás fókuszát.

A vállalatok számára rendkívül fontos a gépi fordítás javítására szánt utómunka mennyisége, a szükséges javítások és az utómunkára ráfordított idő mértékeinek ismerete. Érdemes lenne olyan egynyelvű jegyeket kutatni, amelyek segítségével ezeket tudnánk pontosan megbecsülni.

7.6. Összegzés

Létrehoztam egy egynyelvű minőségbecslő rendszert, amely jól alkalmazható a korpusznyelvészetben vagy a természetesnyelvi elemző rendszerek előfeldolgozó moduljában.

A kutatásból azt a következtetést lehet levonni, hogy az emberek által létrehozott egynyelvű szövegek – a gépi fordítók által generáltakkal ellentétben – nagyobbrészt nem nyelvtani hibákat tartalmaznak. Ezen szövegek minőségi problémái sokkal inkább az internetezők írási szokásaiból adódnak, mint például az ékezetek vagy az írásjelek elhagyása.

További általános jellegű észrevételem, hogy a jegykészlet-optimalizáció nagy jelentőséggel bír. Az eredmények szerint a jegykészlet csökkentésével javítható a teljesítmény, és az erőforrás-felhasználás is kevesebb lesz.

A minőségbecslő módszerének alkalmazásával létrehoztam egy rendszert, amely egynyelvű szövegek minőségét és hibáit tudja megbecsülni.

DOI:10.15774/PPKE.ITK.2019.003

Kapcsolódó tézis

5. tézis: **Létrehoztam egy minőségbecslő rendszert, amellyel egynyelvű szövegek minőségét és hibatípusát lehet meghatározni. A rendszer létrehozásához a gépi fordítás minőségbecsléséhez használt módszert alkalmaztam.**

A tézishez kapcsolódó publikációk: [1] [6] [11] [13].

DOI:10.15774/PPKE.ITK.2019.003

8. fejezet

Összegzés - Új tudományos eredmények

A minőségbecslés témakörben három fő kutatást végeztem. Az első kutatásban létrehoztam egy angol-magyar minőségbecslő rendszert. A rendszer tanításához létrehoztam egy tanítókorpuszt, valamint 27 darab új szemantikai jegyet. A második kutatásomban a gépi fordítórendszerek kimenetének kombinálására a minőségbecslés módszerét használtam fel. Létrehoztam egy kompozit rendszert, amellyel minőségbecslés módszerével különböző gépi fordítórendszer kimenetét kombinálva jobb eredményt értem el, mint a kombinált fordítórendszerek önmagukban. Végül a harmadik kutatásban a minőségbecslés módszerét alkalmaztam, egynyelvű szövegek minőségének becslésére és hibáinak detektálására.

8.1. Angol-magyar minőségbecslés

A minőségbecslés módszere gépi tanuláson alapszik. A modell, jegyek segítségével, a forrásnyelvi és a gép által lefordított mondatokból különböző nyelvfüggetlen és nyelvspecifikus minőségi mutatószámokat nyer ki. Majd a mutatószámok gépi tanuló algoritmussal betanítottam emberi kiértékelésekre.

A modell tanításához tanítókorpuszra lenne szükség, azonban a kutatás ideje alatt nem állt rendelkezésre angol-magyar nyelvű emberi kiértékeléssel rendelkező párhuzamos korpusz. Ezért az angol-magyar minőségbecslő rendszer tanításához létrehoztam egy

DOI:10.15774/PPKE.ITK.2019.003

kézzel kiértékelt tanítókorpuszt. Ennek segítségével létrehoztam egy angol-magyar minőségbecslő rendszert. A felépített rendszeren különböző méréseket végeztem el. Először az angol-spanyol nyelvre optimalizált alapjegykészletet mértem le angol-magyar nyelvre, majd megvizsgáltam a Specia és társai [58] által implementált 76 jegykészletet is. Ezt követően saját szemantikai jegyekkel kísérleteztem. A szemantikai jegyekhez egy angol-magyar szótárat, a WordNetet, a szóbeágyazási modellt és a látens szemantikai analízis módszerét használtam. Végeztem jegykiválasztást is, ami azt jelenti, hogy kevesebb releváns jeggyel sikerült további eredményjavulást elérnem. Ezáltal, kevesebb erőforrással magasabb minőséget értem el.

A WordNet jegyeket angol-spanyol és angol-német nyelvpárokra is kipróbáltam. Mindkét esetben jobb eredményt értem el az alapjegykészlethez képest.

1. tézis: **Létrehoztam egy kézzel kiértékelt korpuszt, amely angol-magyar nyelvű minőségbecslő rendszer tanítására alkalmas.**

2. tézis: **Létrehoztam 27 darab új szemantikai jegyet, kétnyelvű szótárral, a WordNet és a szóbeágyazás módszerével, amelyekkel eredményjavulást értem el az alapjegykészlethez képest.**

3. tézis: **Az általam létrehozott angol-magyar korpusz, valamint a 27 darab új szemantikai jegy segítségével, a QuEst keretrendszer alapján, magyar nyelvtechnológiai eszközök integrálásával, létrehoztam egy minőségbecslő rendszert, angol-magyar nyelvre.**

A tézishez kapcsolódó publikációk: [4] [6] [7] [8] [9] [10].

8.2. Gépi fordítórendszerek kombinálása, minőségbecslés módszerével

A második kutatás során a minőségbecslés módszerét használtam a különböző gépi fordítórendszerek kimeneteinek kombinálására. Az általam létrehozott kompozit rendszer egy kifejezés alapú statisztikai, egy hierarchikus statisztikai és egy neurálishálózat-alapú gépi fordítórendszer kimenetét kombinálja. A rendszer a minőségbecslés módszerével kiválasztja a három rendszer fordításából a legjobb fordítást, és az lesz a rendszer végső kimenete. A módszeremet négy különböző nyelvpárra teszteltem: angol-magyar, angol-német, angol-olasz és angol-japán. Az eredmények alapján rendszerszinten a kompozit

DOI:10.15774/PPKE.ITK.2019.003

gépi fordítórendszerem minden esetben jobb minőséget eredményezett, mint az általa felhasznált rendszerek önmagukban. Angol-magyar nyelvpár esetében nyelvfüggő jegyekkel tovább tudtam növelni a rendszer minőségét.

4. tézis: **Létrehoztam egy kompozit gépi fordítórendszert, amely a minőségbecslés módszerével, különböző gépi fordítórendszerek kimenetét kombinálva ért el rendszerszinten jobb eredményt, mint az általa felhasznált gépi fordítórendszerek önmagukban.**

A tézishez kapcsolódó publikációk: [5] [6] [12].

8.3. Egynyelvű szövegek minőségének és hibáinak meghatározása, minőségbecslés módszerével

A harmadik kutatásban a minőségbecslés módszerét kiterjesztettem egynyelvű szövegek minőségének becslésére. A kutatás célja az volt, hogy megvizsgáljam az interneten elérhető, emberek által produkált szövegek hibáit, valamint a minőségbecslés módszerével, létrehozzak egy automatikus hibadetektáló programot.

Az kutatásom arra mutatott rá, hogy az emberek által létrehozott egynyelvű szövegek, a gépi fordítók által generáltakkal ellentétben, nagyrészt nem nyelvtani hibákat tartalmaznak. Ezen szövegek minőségi problémái inkább az internetezők írási szokásaiból adódnak, mint például az ékezetek vagy az írásjelek elhagyása.

Az általam létrehozott, egynyelvű minőségbecslő rendszer jól alkalmazható a korpusznyelvészetben vagy a természetesnyelvi elemző rendszerek előfeldolgozó moduljaiban.

5. tézis: **Létrehoztam egy minőségbecslő rendszert, amellyel egynyelvű szövegek minőségét és hibatípusát lehet meghatározni. A rendszer létrehozásához a gépi fordítás minőségbecsléséhez használt módszert alkalmaztam.**

A tézishez kapcsolódó publikációk: [1] [6] [11] [13].

DOI:10.15774/PPKE.ITK.2019.003

9. fejezet

A szerző publikációi

Folyóiratcikkek

[1] **Zijian Győző Yang** and L. J. Laki, "πRate: A Task-oriented Monolingual Quality Estimation System", *International Journal of Computational Linguistics and Applications*, 2017, [Megjelenés alatt].

[2] A. Dömötör and **Zijian Győző Yang**, "What's your style? Automatic genre identification with neural network", *International Journal of Computational Linguistics and Applications*, 2018, [Megjelenés alatt].

[3] **Zijian Győző Yang**, "A Gépi fordítás és a neurális gépi fordítás", *Modern Nyelvoktatás*, vol. 24, no. 2–3, pp. 129–139, 2018.

Könyvfejezetek

[4] **Zijian Győző Yang**, L. J. Laki, and B. Siklósi, "Quality Estimation for English-Hungarian with Optimized Semantic Features", in *Computational Linguistics and Intelligent Text Processing*, Konya, Turkey, 2016.

[5] L. J. Laki and **Zijian Győző Yang**, "Combining Machine Translation Systems with Quality Estimation", in *Computational Linguistics and Intelligent Text Processing*, Budapest, Hungary: Springer International Publishing, 2017, pp. 435–444, ISBN: 978-3-319-77116-8.

DOI:10.15774/PPKE.ITK.2019.003

[6] **Zijian Győző Yang**, A. Dömötör, and L. J. Laki, "A Quality Estimation System for Hungarian", in *Human Language Technology. Challenges for Computer Science and Linguistic*, Poznań, Poland: Springer International Publishing, 2018, pp. 201–213, ISBN: 978-3-319-93782-3.

Külföldi konferenciakötetek

[7] **Zijian Győző Yang** and L. J. Laki, "Quality Estimation for English-Hungarian Machine Translation", in *7th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, Poznań, Poland: Uniwersytet im. Adama Mickiewicza w Poznaniu, 2015, pp. 170–174.

[8] **Zijian Győző Yang**, L. J. Laki, and B. Siklósi, "HuQ: An English-Hungarian Corpus for Quality Estimation", in *Proceedings of the LREC 2016 Workshop - Translation Evaluation: From Fragmented Tools and Data Sets to an Integrated Ecosystem*, (May 24, 2016), Portorož, Slovenia, 2016.

Hazai konferenciakötetek

[9] **Zijian Győző Yang**, L. J. Laki, and G. Prószéky, "Gépi fordítás minőségének becslése referencia nélküli módszerrel", in *XI. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2015, pp. 3–13.

[10] **Zijian Győző Yang** and L. J. Laki, "Gépi fordítás minőségbecslésének optimalizálása kétnyelvű szótár és WordNet segítségével", in *XII. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2016, pp. 37–46.

[11] **Zijian Győző Yang** and L. J. Laki, "Minőségbecslő rendszer egynyelvű természetes nyelvi elemzőhöz", in *XIII. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2017, pp. 37–49.

DOI:10.15774/PPKE.ITK.2019.003

[12] L. J. Laki and **Zijian Győző Yang**, “Gépi fordító rendszerek kombinálása minőségbecslés segítségével”, in *XIV. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2018, pp. 281–291.

[13] A. Dömötör and **Zijian Győző Yang**, “Így írtok ti - Nem sztenderd szövegek hibatípusainak detektálása gépi tanulásos módszerrel”, in *XIV. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2018, pp. 305–316.

[14] **Zijian Győző Yang**, “A gépi fordítás kiértékelése”, in *Fókuszban a fordítás értékelése*, Budapest, Hungary: Budapesti Műszaki és Gazdaságtudományi Egyetem Gazdaság - és Társadalom tudományi Kar Idegen Nyelvi Központ, 2018, pp. 147–162.

DOI:10.15774/PPKE.ITK.2019.003

10. fejezet

Irodalomjegyzék

[1] **Zijian Győző Yang** and L. J. Laki, "πRate: A Task-oriented Monolingual Quality Estimation System", *International Journal of Computational Linguistics and Applications*, 2017, [Megjelenés alatt].

[2] A. Dömötör and **Zijian Győző Yang**, "What's your style? Automatic genre identification with neural network", *International Journal of Computational Linguistics and Applications*, 2018, [Megjelenés alatt].

[3] **Zijian Győző Yang**, "A Gépi fordítás és a neurális gépi fordítás", *Modern Nyelvoktatás*, vol. 24, no. 2–3, pp. 129–139, 2018.

[4] **Zijian Győző Yang**, L. J. Laki, and B. Siklósi, "Quality Estimation for English-Hungarian with Optimized Semantic Features", in *Computational Linguistics and Intelligent Text Processing*, Konya, Turkey, 2016.

[5] L. J. Laki and **Zijian Győző Yang**, "Combining Machine Translation Systems with Quality Estimation", in *Computational Linguistics and Intelligent Text Processing*, Budapest, Hungary: Springer International Publishing, 2017, pp. 435–444, ISBN: 978-3-319-77116-8.

[6] **Zijian Győző Yang**, A. Dömötör, and L. J. Laki, "A Quality Estimation System for Hungarian", in *Human Language Technology. Challenges for Computer Science and Linguistic*, Poznań, Poland: Springer International Publishing, 2018, pp. 201–213, ISBN: 978-3-319-93782-3.

DOI:10.15774/PPKE.ITK.2019.003

[7] **Zijian Győző Yang** and L. J. Laki, "Quality Estimation for English-Hungarian Machine Translation", in *7th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, Poznań, Poland: Uniwersytet im. Adama Mickiewicza w Poznaniu, 2015, pp. 170–174.

[8] **Zijian Győző Yang**, L. J. Laki, and B. Siklósi, "HuQ: An English-Hungarian Corpus for Quality Estimation", in *Proceedings of the LREC 2016 Workshop - Translation Evaluation: From Fragmented Tools and Data Sets to an Integrated Ecosystem*, (May 24, 2016), Portorož, Slovenia, 2016.

[9] **Zijian Győző Yang**, L. J. Laki, and G. Prószéky, "Gépi fordítás minőségének becslése referencia nélküli módszerrel", in *XI. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2015, pp. 3–13.

[10] **Zijian Győző Yang** and L. J. Laki, "Gépi fordítás minőségbecslésének optimalizálása kétnyelvű szótár és WordNet segítségével", in *XII. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2016, pp. 37–46.

[11] **Zijian Győző Yang** and L. J. Laki, "Minőségbecslő rendszer egynyelvű természetes nyelvi elemzőhöz", in *XIII. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2017, pp. 37–49.

[12] L. J. Laki and **Zijian Győző Yang**, "Gépi fordító rendszerek kombinálása minőségbecslés segítségével", in *XIV. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2018, pp. 281–291.

[13] A. Dömötör and **Zijian Győző Yang**, "Így írtok ti - Nem sztenderd szövegek hibatípusainak detektálása gépi tanulásos módszerrel", in *XIV. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Tanszékcsoport, 2018, pp. 305–316.

DOI:10.15774/PPKE.ITK.2019.003

[14] **Zijian Győző Yang**, "A gépi fordítás kiértékelése", in *Fókuszban a fordítás értékelése*, Budapest, Hungary: Budapesti Műszaki és Gazdaságtudományi Egyetem Gazdaság - és Társadalom tudományi Kar Idegen Nyelvi Központ, 2018, pp. 147–162.

[15] W. J. Hutchins and H. L. Somers, *An Introduction to Machine Translation*. London, United Kingdom: Academic Press, 1992, ISBN: 012362830X, 978-0123628305.

[16] P. Koehn, *Statistical Machine Translation*, 1st. New York, USA: Cambridge University Press, 2010, p. 219, ISBN: 0521874157, 9780521874151.

[17] H. Somers, "Review Article: Example-based Machine Translation", *Machine Translation*, vol. 14, pp. 113–157, 1999.

[18] C. E. Shannon, "A Mathematical Theory of Communication", *Bell System Technical Journal*, vol. 27, pp. 379–423, 1948.

[19] A. Lopez, "Statistical Machine Translation", *ACM Computing Surveys*, vol. 40, no. 3, 8:1–8:49, 2008, ISSN: 0360-0300.

[20] D. Chiang, "A Hierarchical Phrase-based Model for Statistical Machine Translation", in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, Ann Arbor, Michigan: Association for Computational Linguistics, 2005, pp. 263–270.

[21] L. J. Laki, A. Novák, and B. Siklósi, "Syntax Based Reordering in Phrase Based English-Hungarian Statistical Machine Translation", *International Journal of Computational Linguistics and Applications*, vol. 4, 63–78, 2013, ISSN: 0976-0962.

[22] V. Vandeghinste, "Scaling up a Hybrid MT System: From low to full resources", *Linguistica Antverpiensia, New Series – Themes in Translation Studies*, vol. 0, no. 8, 2013, ISSN: 2295-5739.

[23] V. Vandeghinste, *A Hybrid Modular Machine Translation System*. Utrecht, Netherlands: LOT, 2008, ISBN: 978-90-78328-50-6.

[24] W. S. McCulloch and W. Pitts, "A Logical Calculus of the ideas immanent in Nervous Activity", *The bulletin of mathematical biophysics*, vol. 5, pp. 115–133, 1943.

DOI:10.15774/PPKE.ITK.2019.003

[25] P. Tutorials, *Artificial Intelligence*, https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_neural_networks.htm, Online; [Hozzáférés dátuma: 2019.01.10], 2018.

[26] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate", in *Proceedings of International Conference on Learning Representations*, San Diego, CA, USA, 2015.

[27] C. Olah, *Understanding LSTM Networks*, http://colah.github.io/posts/2015-08-Understanding-LSTMs, Online; [Hozzáférés dátuma: 2019.01.13], 2015.

[28] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks", in *Advances in Neural Information Processing Systems 27*, Montréal, Canada: Curran Associates, Inc., 2014, pp. 3104–3112.

[29] K. Cho, B. v. Merrienboer, Gülçehre, Çaglar, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation", in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, Doha, Qatar, 2014, pp. 1724–1734.

[30] N. Kalchbrenner and P. Blunsom, "Recurrent Continuous Translation Models", *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013.

[31] S. Merity, *Peeking into the neural network architecture used for Google's Neural Machine Translation*, https://smerity.com/articles/2016/google_nmt_arch.html, Online; [Hozzáférés dátuma: 2019.01.10.], 2016.

[32] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality", in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems*, Nevada, United States, 2013, pp. 3111–3119.

[33] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space", in *International Conference on Learning Representations*, Scottsdale, USA, 2013.

DOI:10.15774/PPKE.ITK.2019.003

[34] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011, ISBN: 0123748569, 9780123748560.

[35] G. W. Flake and S. Lawrence, "Efficient SVM Regression Training with SMO", *Machine Learning*, vol. 46, no. 1, pp. 271–290, 2002, ISSN: 1573-0565.

[36] D. J. Mackay, "Introduction to Gaussian Processes", in *Neural Networks and Machine Learning*, Cambridge, UK, 1998.

[37] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. London, England: The MIT Press, 2006, p. 248, ISBN: 026218253X.

[38] M. A. Hall, "Correlation-based Feature Subset Selection for Machine Learning", PhD thesis, University of Waikato, Hamilton, New Zealand, 1998.

[39] M. Miháltz, C. Hatvani, J. Kuti, G. Szarvas, J. Csirik, G. Prószéky, and T. Váradi, "Methods and Results of the Hungarian WordNet Project", in *Proceedings of the Fourth Global WordNet Conference GWC 2008*, 2008, pp. 310–320.

[40] I. Niles and A. Pease, "Towards a Standard Upper Ontology", in *Proceedings of the International Conference on Formal Ontology in Information Systems - Volume 2001*, ser. FOIS '01, Ogunquit, Maine, USA: ACM, 2001, pp. 2–9, ISBN: 1-58113-377-4.

[41] C. Fellbaum, *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.

[42] D. Jurafsky and J. H. Martin, *Speech and Language Processing (2Nd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2009, ISBN: 0131873210.

[43] D. Langlois, "LORIA System for the WMT15 Quality Estimation Shared Task", in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, Lisbon, Portugal: Association for Computational Linguistics, 2015, pp. 323–329.

[44] K. Gwet, *Handbook of Inter-Rater Reliability: The Definitive Guide to Measuring the Extent of Agreement Among Raters*, 4rd. Gaithersburg, USA: Advanced Analytics LLC, 2014, ISBN: 9780970806284.

DOI:10.15774/PPKE.ITK.2019.003

[45] S. Banerjee and A. Lavie, “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments”, in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, Ann Arbor, Michigan: Association for Computational Linguistics, 2005, pp. 65–72.

[46] A. Varga, “A gépi fordítás minősége és javítási lehetőségei”, PhD dissertation, Eötvös Lóránd Tudományegyetem, 2011.

[47] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: A Method for Automatic Evaluation of Machine Translation”, in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, Philadelphia, Pennsylvania: Association for Computational Linguistics, 2002, pp. 311–318.

[48] FTSK, *OrthoBLEU – MT Evalution Based on Orthographic Similarities*, http://www.fask.uni-mainz.de/user/rapp/comtrans/d05orthobleu.html, Online; [Hozzáférés dátuma: 2019.01.10.], 2014.

[49] C.-Y. Lin and F. J. Och, “Automatic Evaluation of Machine Translation Quality Using Longest Common Subsequence and Skip-bigram Statistics”, in *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ser. ACL ’04, Barcelona, Spain: Association for Computational Linguistics, 2004, pp. 605–612.

[50] C.-Y. Lin, “ROUGE: A Package for Automatic Evaluation of summaries”, in *Proceedings of the ACL-04 Workshop on Text Summarization Branches Out*, Barcelona, Spain, 2004, pp. 74–82.

[51] A. Lavie and A. Agarwal, “METEOR: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments”, in *Proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic: Association for Computational Linguistics, 2007, pp. 228–231.

[52] A. L.-F. Han, D. Wong, and L. S Chao, “LEPOR: A Robust Evaluation Metric for Machine Translation with Augmented Factors”, in *Proceedings of the 24th International Conference on Computational Linguistics*, Mumbai, India, Dec. 2012, pp. 441–450.

DOI:10.15774/PPKE.ITK.2019.003

[53] H. Isozakim, T. Hirao, K. Duh, K. Sudoh, and H. Tsukada, “Automatic Evaluation of Translation Quality for Distant Language Pairs”, in *Conference on Empirical Methods on Natural Language Processing*, Massachusetts, USA, 2010, 944—952.

[54] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, “A study of translation edit rate with targeted human annotation”, in *In Proceedings of Association for Machine Translation in the Americas*, Cambridge, USA, 2006, pp. 223–231.

[55] M. Gamon, A. Aue, and M. Smets, “Sentence-Level MT Evaluation Without Reference Translations: Beyond Language Modeling”, in *EAMT 2005 Conference Proceedings*, Springer-Verlag, 2005, 103—111.

[56] J. Albrecht and R. Hwa, “Regression for Sentence-Level MT Evaluation with Pseudo References”, in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, Prague, Czech Republic, 2007, 296—303.

[57] L. Specia, M. Turchi, N. Cancedda, M. Dymetman, and C. N., “Estimating the Sentence-Level Quality of Machine Translation Systems”, in *Proceedings of the. 13th Annual Conference. of the European Association. for Machine Translation*, Barcelona, Spain, 2009, pp. 28–37.

[58] L. Specia, K. Shah, J. G. de Souza, and T. Cohn, “QuEst - A translation quality estimation framework”, in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Sofia, Bulgaria, 2013, pp. 79–84.

[59] H. Kim and J.-H. Lee, “Recurrent Neural Network based Translation Quality Estimation”, in *Proceedings of the First Conference on Machine Translation*, Berlin, Germany, 2016, pp. 787–792.

[60] J. Ive, F. Blain, and L. Specia, “deepQuest: A Framework for Neural-based Quality Estimation”, in *Proceedings of the 27th International Conference on Computational Linguistics*, New Mexico, USA, 2018, pp. 3146–3157.

[61] T. Luong, H. Pham, and C. D. Manning, “Effective Approaches to Attention-based Neural Machine Translation”, in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal: Association for Computational Linguistics, 2015, pp. 1412–1421.

DOI:10.15774/PPKE.ITK.2019.003

[62] J. Wang, K. Fan, B. Li, F. Zhou, B. Chen, Y. Shi, and L. Si, "Alibaba Submission for WMT18 Quality Estimation Task", in *Proceedings of the Third Conference on Machine Translation*, Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 809–815.

[63] A. Vaswani, N. Shazeer, N. Parmar, U. Jakob, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need", in *31st Conference on Neural Information Processing System*, California, USA, 2017.

[64] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures", *Neural Networks*, vol. 18, no. 5, pp. 602–610, 2005, ISSN: 0893-6080.

[65] N. A. C. of the Association for Computational Linguistics, *Shared Task: Quality Estimation*, http://www.statmt.org/wmt12/quality-estimation-task.html, Online; [Hozzáférés dátuma: 2019.01.13], 2012.

[66] E. M. in Natural Language Processing, *Shared Task: Quality Estimation*, http://www.statmt.org/wmt18/quality-estimation-task.html, Online; [Hozzáférés dátuma: 2019.01.13], 2018.

[72] N. A. C. of the Association for Computational Linguistics, *Shared Task: Quality Estimation*, http://www.statmt.org/wmt13/quality-estimation-task.html, Online; [Hozzáférés dátuma: 2019.01.13], 2013.

[73] ——, *Shared Task: Quality Estimation*, http://www.statmt.org/wmt14/quality-estimation-task.html, Online; [Hozzáférés dátuma: 2019.01.13], 2014.

[67] D. Varga, P. Halácsy, A. Kornai, V. Nagy, L. Németh, and V. Trón, "Parallel corpora for medium density languages", in *In Proceedings of the RANLP*, Borovets, Bulgaria: INCOMA Ltd., 2005, pp. 590–596.

[68] A. Novák, L. Tihanyi, and G. Prószéky, "The MetaMorpho Translation System", in *Proceedings of the Third Workshop on Statistical Machine Translation*, ser. StatMT '08, Columbus, Ohio, 2008, pp. 111–114, ISBN: 978-1-932432-09-1.

DOI:10.15774/PPKE.ITK.2019.003

[69] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open Source Toolkit for Statistical Machine Translation", in *Proceedings of the 45th Annual Meeting of the ACL*, Prague, Czech Republic, 2007, pp. 177–180.

[70] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, *et al.*, "Google's Neural Machine Translation System: Bridging the gap between human and machine translation", *Technical Report*, 2016.

[71] A. Novák, "Milyen a jó Humor?", in *Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, 2003, pp. 138–145.

[74] L. Specia, G. Paetzold, and C. Scarton, "Multi-level Translation Quality Prediction with QuEst++", in *ACL-IJCNLP 2015 System Demonstrations*, Beijing, China, 2015, pp. 115–120.

[75] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA Data Mining Software: An Update", *SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009, ISSN: 1931-0145.

[76] D. Beck, K. Shah, T. Cohn, and L. Specia, "SHEF-Lite: When Less is More for Translation Quality Estimation", in *Proceedings of the Workshop on Machine Translation*, Sofia, Bulgaria, 2013, pp. 337–342.

[77] O. Bojar, R. Chatterjee, C. Federmann, B. Haddow, M. Huck, C. Hokamp, P. Koehn, V. Logacheva, C. Monz, M. Negri, M. Post, C. Scarton, L. Specia, and M. Turchi, "Findings of the 2015 Workshop on Statistical Machine Translation", in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, Lisbon, Portugal: Association for Computational Linguistics, 2015, pp. 1–46.

[78] O. Bojar, R. Chatterjee, C. Federmann, Y. Graham, B. Haddow, M. Huck, A. Jimeno Yepes, P. Koehn, V. Logacheva, C. Monz, M. Negri, A. Neveol, M. Neves, M. Popel, M. Post, R. Rubino, C. Scarton, L. Specia, M. Turchi, K. Verspoor,

DOI:10.15774/PPKE.ITK.2019.003

and M. Zampieri, “Findings of the 2016 Conference on Machine Translation”, in *Proceedings of the First Conference on Machine Translation*, Berlin, Germany: Association for Computational Linguistics, 2016, pp. 131–198.

[79] O. Bojar, R. Chatterjee, C. Federmann, Y. Graham, B. Haddow, S. Huang, M. Huck, P. Koehn, Q. Liu, V. Logacheva, C. Monz, M. Negri, M. Post, R. Rubino, L. Specia, and M. Turchi, “Findings of the 2017 Conference on Machine Translation (WMT17)”, in *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, Copenhagen, Denmark: Association for Computational Linguistics, 2017, pp. 169–214.

[88] E. Biçici, “Feature Decay Algorithms for Fast Deployment of Accurate Statistical Machine Translation Systems”, in *Proceedings of the Eighth Workshop on Statistical Machine Translation*, Sofia, Bulgaria, 2013, pp. 78–84.

[89] J. G. Camargo de Souza, C. Buck, M. Turchi, and M. Negri, “FBK-UEdin Participation to the WMT13 Quality Estimation Shared Task”, in *Proceedings of the Eighth Workshop on Statistical Machine Translation*, Sofia, Bulgaria, 2013, pp. 352–358.

[90] N. Q. Luong, B. Lecouteux, and L. Besacier, “LIG System for WMT13 QE Task: Investigating the Usefulness of Features in Word Confidence Estimation for MT”, in *Proceedings of the Eighth Workshop on Statistical Machine Translation*, Sofia, Bulgaria: Association for Computational Linguistics, 2013, pp. 386–391.

[80] G. Orosz and A. Novák, “PurePos 2.0: a hybrid tool for morphological disambiguation”, in *Proceedings of Recent Advances in Natural Language Processing*, Hissar, Bulgaria, 2013, pp. 539–545.

[81] G. Prószéky, “Industrial Applications of Unification Morphology”, in *Proceedings of the Fourth Conference on ANLP*, Stuttgart, Germany, 1994, pp. 213–214.

[82] G. Recski and D. Varga, “A Hungarian NP Chunker”, *The Odd Yearbook*, pp. 87–93, 2009.

[83] D. Csendes, J. Csirik, T. Gyimóthy, and A. Kocsor, “The Szeged Treebank”, in *Lecture Notes in Computer Science: Text, Speech and Dialogue*, Karlovy Vary, Czech Republic, 2005, pp. 123–131.

DOI:10.15774/PPKE.ITK.2019.003

[84] A. Novák and B. Novák, “Magyar szóbeágyazási modellek kézi kiértékelése”, in *XIV. Magyar Számítógépes Nyelvészeti Konferencia*, Szeged, Hungary: Szegedi Tudományegyetem, 2018.

[85] S. L. Salzberg, “C4.5: Programs for Machine Learning”, *Machine Learning*, vol. 16, no. 3, pp. 235–240, 1994, ISSN: 1573-0565.

[86] A. Gonzalez-Agirre, E. Laparra, and G. Rigau, “Multilingual Central Repository version 3.0.”, in *LREC*, Rigau, German: European Language Resources Association (ELRA), 2012, pp. 2525–2529, ISBN: 978-2-9517408-7-7.

[87] G. de Melo and G. Weikum, “Towards a Universal Wordnet by Learning from Combined Evidence”, in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, ser. CIKM ’09, Hong Kong, China: ACM, 2009, pp. 513–522, ISBN: 978-1-60558-512-3. DOI: 10.1145/1645953.1646020.

[91] O. Bojar, C. Federmann, M. Fishel, Y. Graham, B. Haddow, M. Huck, P. Koehn, and C. Monz, “Findings of the 2018 Conference on Machine Translation (WMT18)”, in *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*, Belgium, Brussels: Association for Computational Linguistics, 2018, pp. 272–307.

[100] F. Huang and K. Papineni, “Hierarchical System Combination for Machine Translation”, in *EMNLP-CoNLL*, Prague, Czech Republic, 2007, pp. 277–286.

[93] E. Matusov, N. Ueffing, and H. Ney, “Computing Consensus Translation for Multiple Machine Translation Systems Using Enhanced Hypothesis Alignment”, in *11st Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference*, Trento, Italy: The Association for Computer Linguistics, 2006, pp. 33–40, ISBN: 1-932432-59-0.

[101] K. C. Sim, W. J. Byrne, M. J. F Gales, H. Sahbi, and P. C. Woodland, “Consensus Network Decoding for Statistical Machine Translation System Combination”, in *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, 2007, pp. IV–105–IV–108.

DOI:10.15774/PPKE.ITK.2019.003

[102] A.-V. Rosti, B. Zhang, S. Matsoukas, and R. Schwartz, “Incremental Hypothesis Alignment for Building Confusion Networks with Application to Machine Translation System Combination”, in *Proceedings of the Third Workshop on Statistical Machine Translation*, Columbus, Ohio: Association for Computational Linguistics, 2008, pp. 183–186.

[103] K. Heafield, G. Hanneman, and A. Lavie, “Machine Translation System Combination with Flexible Word Ordering”, in *Proceedings of the Fourth Workshop on Statistical Machine Translation*, Athens, Greece: Association for Computational Linguistics, 2009, pp. 56–60.

[104] T. Okita and J. van Genabith, “Minimum Bayes Risk Decoding with Enlarged Hypothesis Space in System Combination”, in *Computational Linguistics and Intelligent Text Processing: 13th International Conference*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 40–51.

[94] T. Okita, R. Rubino, and J. v. Genabith, “Sentence-Level Quality Estimation for MT System Combination”, in *Proceedings of the Second Workshop on Applying Machine Learning Techniques to Optimise the Division of Labour in Hybrid MT*, Mumbai, India: The COLING 2012 Organizing Committee, 2012, pp. 55–64.

[95] F. J. Och and H. Ney, “A Systematic Comparison of Various Statistical Alignment Models”, *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[96] M. Federico, N. Bertoldi, and M. Cettolo, “IRSTLM: an open source toolkit for handling large scale language models”, in *INTERSPEECH 2008*, Brisbane, Australia, 2008, pp. 1618–1621.

[97] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. Rush, “OpenNMT: Open-Source Toolkit for Neural Machine Translation”, in *Proceedings of ACL 2017, System Demonstrations*, Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 67–72.

[98] D. Bahdanau, K. Cho, and Y. Bengio, “Neural Machine Translation by Jointly Learning to Align and Translate”, in *6th International Conference on Learning Representations*, San Diego, USA, 2015.

DOI:10.15774/PPKE.ITK.2019.003

[99] S. Mei, A. Montanari, and P.-M. Nguyen, “A mean field view of the landscape of two-layer neural networks”, *Proceedings of the National Academy of Sciences*, vol. 115, no. 33, E7665–E7671, 2018, ISSN: 0027-8424.

[105] C. Oravecz, T. Váradi, and B. Sass, “The Hungarian Gigaword Corpus”, in *Proceedings of the 9th International Conference on Language Resources and Evaluation*, Reykjavik, Iceland: ELRA, 2014, ISBN: 978-2-9517408-8-4.

[106] I. Endrédy and G. Prószéky, “A Pázmány Korpusz”, in *Nyelvtudományi Közlemények*, 2016, pp. 191–206.

[107] J. Read, P. Reutemann, B. Pfahringer, and G. Holmes, “MEKA: A Multi-label/Multi-target Extension to Weka”, *Journal of Machine Learning Research*, vol. 17, no. 21, pp. 1–5, 2016.

[108] J. Read, B. Pfahringer, G. Holmes, and E. Frank, “Classifier Chains for Multi-label Classification”, *Machine Learning*, vol. 85, no. 3, pp. 333–359, 2011, ISSN: 0885-6125.

[109] H. Bechara, C. P. Escartin, C. Orasan, and L. Specia, “Semantic Textual Similarity in Quality Estimation”, *Baltic Journal of Modern Computing*, vol. 4, no. 2, pp. 256–268, 2016.

DOI:10.15774/PPKE.ITK.2019.003

A. függelék

Az angol-magyar nyelvű minőségbecsléshez felhasznált jegyek

A jelen függelékben található az angol-magyar minőségbecslő rendszer kutatásához kísérletezett és felhasznált jegyek összessége.

A.1. Felhasznált black-box jegyek

Az A.1. táblázatban található az összes általam felhasznált 76 black-box jegy, amelyeket Lucia és társai [58] implementáltak.

Azonosító	Leírás
1001	Tokenek száma a forrásmondatban.
1002	Tokenek száma a célmondatban.
1003	Tokenek aránya a forrás- és a célmondatban.
1004	Tokenek száma a célmondatban / Tokenek száma a forrásmondatban.
1005	Tokenek számának abszolút értékben vett különbsége a forrás- és a célmondatban, a forrásmondat hosszával normalizálva.
1006	Tokenek átlagos hossza a forrásmondatban.
1007	Hibás zárójelek száma.
1008	Hibás idézőjelek száma.

DOI:10.15774/PPKE.ITK.2019.003

1009	Forrásmondat nyelvmodell valószínűsége.
1010	Forrásmondat perplexitása.
1011	Forrásmondat perplexitása mondatvégi írásjel nélkül.
1012	Célmondat nyelvmodell valószínűsége.
1013	Célmondat perplexitása.
1014	Célmondat perplexitása mondatvégi írásjel nélkül.
1015	A célnyelvi szó előfordulásának száma a célnyelvi hipotézisben.
1016	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}01$).
1018	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}05$).
1020	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}1$).
1022	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}2$).
1024	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}5$).
1026	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}01$), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.
1028	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}05$), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.
1030	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}1$), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.
1032	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}2$), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.

DOI:10.15774/PPKE.ITK.2019.003

1034 Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,5), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.

1036 Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,01), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.

1038 Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,05), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.

1040 Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,1), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.

1042 Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,2), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.

1044 Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,5), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.

1046 Forrásnyelvi átlagos unigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1047 Forrásnyelvi átlagos unigram gyakoriság a második kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1048 Forrásnyelvi átlagos unigram gyakoriság a harmadik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1049 Forrásnyelvi átlagos unigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1050 Forrásnyelvi átlagos bigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1051 Forrásnyelvi átlagos bigram gyakoriság a második kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

DOI:10.15774/PPKE.ITK.2019.003

1052 Forrásnyelvi átlagos bigram gyakoriság a harmadik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1053 Forrásnyelvi átlagos bigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1054 Forrásnyelvi átlagos trigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1055 Forrásnyelvi átlagos trigram gyakoriság a második kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1056 Forrásnyelvi átlagos trigram gyakoriság a harmadik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1057 Forrásnyelvi átlagos trigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1058 Forrásnyelvi korpuszban lévő különböző unigramok aránya (minden kvartilisben).

1059 Forrásnyelvi korpuszban lévő különböző bigramok aránya (minden kvartilisben).

1060 Forrásnyelvi korpuszban lévő különböző trigramok aránya (minden kvartilisben).

1061 Átlagos szógyakoriság: forrásmondatban lévő minden type (unigram), ami a x-szer feltűnik a korpuszban (minden kvartilisben).

1062 A forrás- és a célmondatban lévő pontok számának abszolút értékben vett különbsége.

1063 A forrás- és a célmondatban lévő pontok számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

1064 A forrás- és a célmondatban lévő vesszők számának abszolút értékben vett különbsége.

1065 A forrás- és a célmondatban lévő vesszők számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

1066 A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége.

DOI:10.15774/PPKE.ITK.2019.003

1067	A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1068	A forrás- és a célmondatban lévő pontosvesszők számának abszolút értékben vett különbsége.
1069	A forrás- és a célmondatban lévő pontosvesszők számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1070	A forrás- és a célmondatban lévő kérdőjelek számának abszolút értékben vett különbsége.
1071	A forrás- és a célmondatban lévő kérdőjelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1072	A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége.
1073	A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1074	Írásjelek száma a forrásmondatban.
1075	Írásjelek száma a célmondatban.
1076	A forrás- és a célmondatban lévő írásjelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1077	Számok aránya a forrásmondatban.
1078	Számok aránya a célmondatban.
1079	A forrás- és a célmondatban lévő számok számának abszolút értékben vett különbsége, a forrásmondat hosszával normalizálva.
1080	Tokenek száma a forrásmondatban, amelyek nem csak a-z betűt tartalmaznak.
1081	Tokenek aránya a célmondatban, amelyek nem csak a-z betűt tartalmaznak.
1082	A forrás- és a célmondatban lévő csak a-z betűt tartalmazó tokenek aránya.
1088	Főnevek aránya a forrásmondatban.
1089	Igék aránya a forrásmondatban.
1090	Főnevek aránya a célmondatban.
1091	Igék aránya a célmondatban.
1092	Főnevek aránya a forrás- és a célmondatban.

DOI:10.15774/PPKE.ITK.2019.003

1093	Igék aránya a forrás- és a célmondatban.
1094	Névmások aránya a forrás- és a célmondatban.
2004	A forrás- és a célmondatban lévő NP-k számának abszolút értékben vett különbsége.
2005	A forrás- és a célmondatban lévő NP-k számának abszolút értékben vett különbsége, a kifejezési címkék számával normalizálva.

A.1. táblázat Hun-Quest black-box jegyei

A.2. Alapjegykészlet

Az A.2. táblázatban található a 17 jegyből álló alapjegykészlet (baseline), amelyeket Lucia és társai [58] implementáltak.

Azonosító	**Leírás**
1001	Tokenek száma a forrásmondatban.
1002	Tokenek száma a célmondatban.
1006	Tokenek átlagos hossza a forrásmondatban.
1009	Forrásmondat nyelvmodell valószínűsége.
1012	Célmondat nyelvmodell valószínűsége.
1015	A célnyelvi szó előfordulásának száma a célnyelvi hipotézisben.
1022	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $>$ 0,2).
1036	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $>$ 0,01), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
1046	Forrásnyelvi átlagos unigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1049	Forrásnyelvi átlagos unigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1050	Forrásnyelvi átlagos bigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

DOI:10.15774/PPKE.ITK.2019.003

1053	Forrásnyelvi átlagos bigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1054	Forrásnyelvi átlagos trigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1057	Forrásnyelvi átlagos trigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1058	Forrásnyelvi korpuszban lévő különböző unigramok aránya (minden kvartilisben).
1074	Írásjelek száma a forrásmondatban.
1075	Írásjelek száma a célmondatban.

A.2. táblázat Alapjegykészlet

A.3. Szemantikai jegyek

Az A.3. táblázatban található az általam létrehozott 3 szótári jegy és a 72 WordNet jegy szemantikai jegy.

Azonosító	**Leírás**
2001	Szótári illeszkedés a célmondatban.
2002	Szótári illeszkedés a forrásmondatban.
2003	Szótári illeszkedés F-mértéke.
2006	WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2007	WordNet illeszkedés F-mértéke: igék illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2008	WordNet illeszkedés F-mértéke: melléknevek illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2009	WordNet illeszkedés F-mértéke: határozószók illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2010	WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / főnevek száma (+szóbeágyazás)

DOI:10.15774/PPKE.ITK.2019.003

2011	WordNet illeszkedés F-mértéke: igék illeszkedésének száma / igék száma. (+szóbeágyazás)
2012	WordNet illeszkedés F-mértéke: melléknevek illeszkedésének száma / melléknevek száma. (+szóbeágyazás)
2013	WordNet illeszkedés F-mértéke: határozószók illeszkedésének száma / határozószók száma. (+szóbeágyazás)
2014	WordNet illeszkedés a célmondatban: főnevek illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2015	WordNet illeszkedés a célmondatban: igék illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2016	WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2017	WordNet illeszkedés a célmondatban: határozószók illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2018	WordNet illeszkedés a célmondatban: főnevek illeszkedésének száma / főnevek száma. (+szóbeágyazás)
2019	WordNet illeszkedés a célmondatban: igék illeszkedésének száma / igék száma. (+szóbeágyazás)
2020	WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / melléknevek száma. (+szóbeágyazás)
2021	WordNet illeszkedés a célmondatban: határozószók illeszkedésének száma / határozószók száma. (+szóbeágyazás)
2022	WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2023	WordNet illeszkedés a forrásmondatban: igék illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2024	WordNet illeszkedés a forrásmondatban: melléknevek illeszkedésének száma / tokenek száma. (+szóbeágyazás)
2025	WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / tokenek száma. (+szóbeágyazás)

DOI:10.15774/PPKE.ITK.2019.003

2026	WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / főnevek száma. (+szóbeágyazás)
2027	WordNet illeszkedés a forrásmondatban: igék illeszkedésének száma / igék száma. (+szóbeágyazás)
2028	WordNet illeszkedés a forrásmondatban: melléknevek illeszkedésének száma / melléknevek száma. (+szóbeágyazás)
2029	WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / határozószók száma. (+szóbeágyazás)
2006a	WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / tokenek száma.
2007a	WordNet illeszkedés F-mértéke: igék illeszkedésének száma / tokenek száma.
2008a	WordNet illeszkedés F-mértéke: melléknevek illeszkedésének száma / tokenek száma.
2009a	WordNet illeszkedés F-mértéke: határozószók illeszkedésének száma / tokenek száma.
2010a	WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / főnevek száma
2011a	WordNet illeszkedés F-mértéke: igék illeszkedésének száma / igék száma.
2012a	WordNet illeszkedés F-mértéke: melléknevek illeszkedésének száma / melléknevek száma.
2013a	WordNet illeszkedés F-mértéke: határozószók illeszkedésének száma / határozószók száma.
2014a	WordNet illeszkedés a célmondatban: főnevek illeszkedésének száma / tokenek száma.
2015a	WordNet illeszkedés a célmondatban: igék illeszkedésének száma / tokenek száma.
2016a	WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / tokenek száma.

DOI:10.15774/PPKE.ITK.2019.003

2017a	WordNet illeszkedés a célmondatban: határozószók illeszkedésének száma / tokenek száma.
2018a	WordNet illeszkedés a célmondatban: főnevek illeszkedésének száma / főnevek száma.
2019a	WordNet illeszkedés a célmondatban: igék illeszkedésének száma / igék száma.
2020a	WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / melléknevek száma.
2021a	WordNet illeszkedés a célmondatban: határozószók illeszkedésének száma / határozószók száma.
2022a	WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / tokenek száma.
2023a	WordNet illeszkedés a forrásmondatban: igék illeszkedésének száma / tokenek száma.
2024a	WordNet illeszkedés a forrásmondatban: melléknevek illeszkedésének száma / tokenek száma.
2025a	WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / tokenek száma.
2026a	WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / főnevek száma.
2027a	WordNet illeszkedés a forrásmondatban: igék illeszkedésének száma / igék száma.
2028a	WordNet illeszkedés a forrásmondatban: melléknevek illeszkedésének száma / melléknevek száma.
2029a	WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / határozószók száma.
2006lsa	WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / tokenek száma. (+LSA)
2007lsa	WordNet illeszkedés F-mértéke: igék illeszkedésének száma / tokenek száma. (+LSA)

DOI:10.15774/PPKE.ITK.2019.003

2008lsa	WordNet illeszkedés F-mértéke: melléknevek illeszkedésének száma / tokenek száma. (+LSA)
2009lsa	WordNet illeszkedés F-mértéke: határozószók illeszkedésének száma / tokenek száma. (+LSA)
2010lsa	WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / főnevek száma (+LSA)
2011lsa	WordNet illeszkedés F-mértéke: igék illeszkedésének száma / igék száma. (+LSA)
2012lsa	WordNet illeszkedés F-mértéke: melléknevek illeszkedésének száma / melléknevek száma. (+LSA)
2013lsa	WordNet illeszkedés F-mértéke: határozószók illeszkedésének száma / határozószók száma. (+LSA)
2014lsa	WordNet illeszkedés a célmondatban: főnevek illeszkedésének száma / tokenek száma. (+LSA)
2015lsa	WordNet illeszkedés a célmondatban: igék illeszkedésének száma / tokenek száma. (+LSA)
2016lsa	WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / tokenek száma. (+LSA)
2017lsa	WordNet illeszkedés a célmondatban: határozószók illeszkedésének száma / tokenek száma. (+LSA)
2018lsa	WordNet illeszkedés a célmondatban: főnevek illeszkedésének száma / főnevek száma. (+LSA)
2019lsa	WordNet illeszkedés a célmondatban: igék illeszkedésének száma / igék száma. (+LSA)
2020lsa	WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / melléknevek száma. (+LSA)
2021lsa	WordNet illeszkedés a célmondatban: határozószók illeszkedésének száma / határozószók száma. (+LSA)
2022lsa	WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / tokenek száma. (+LSA)

DOI:10.15774/PPKE.ITK.2019.003

2023lsa	WordNet illeszkedés a forrásmondatban: igék illeszkedésének száma / tokenek száma. (+LSA)
2024lsa	WordNet illeszkedés a forrásmondatban: melléknevek illeszkedésének száma / tokenek száma. (+LSA)
2025lsa	WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / tokenek száma. (+LSA)
2026lsa	WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / főnevek száma. (+LSA)
2027lsa	WordNet illeszkedés a forrásmondatban: igék illeszkedésének száma / igék száma. (+LSA)
2028lsa	WordNet illeszkedés a forrásmondatban: melléknevek illeszkedésének száma / melléknevek száma. (+LSA)
2029lsa	WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / határozószók száma. (+LSA)

A.3. táblázat 75 szemantikai jegy

A.4. Optimalizált jegyek

Az A.4. táblázatban található a tartalomhűség értékekre (TA) betanított minőségbecslő modell optimalizált, 29 jegyből álló jegykészlete (OptTA), a relevancia sorrendjében. A vastagon szedett sorok jelzik az általam készített szemantikai jegyeket.

Azonosító	**Leírás**
1064	A forrás- és a célmondatban lévő vesszők számának abszolút értékben vett különbsége.
1015	A célnyelvi szó előfordulásának száma a célnyelvi hipotézisben.
1091	Igék aránya a célmondatban.
1089	Igék aránya a forrásmondatban.
2005	A forrás- és a célmondatban lévő NP-k számának abszolút értékben vett különbsége, a kifejezési címkék számával normalizálva.
1001	Tokenek száma a forrásmondatban.
1075	Írásjelek száma a célmondatban.

DOI:10.15774/PPKE.ITK.2019.003

1072 A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége.

1057 Forrásnyelvi átlagos trigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1066 A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége.

1024 Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}5$).

1082 A forrás- és a célmondatban lévő csak a-z betűt tartalmazó tokenek aránya.

1042 Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}2$), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.

1094 Névmások aránya a forrás- és a célmondatban.

1010 Forrásmondat perplexitása.

1068 A forrás- és a célmondatban lévő pontosvesszők számának abszolút értékben vett különbsége.

2019 WordNet illeszkedés a célmondatban: igék illeszkedésének száma / igék száma.

1006 Tokenek átlagos hossza a forrásmondatban.

1060 Forrásnyelvi korpuszban lévő különböző trigramok aránya (minden kvartilisben).

1013 Célmondat perplexitása.

2023 WordNet illeszkedés a forrásmondatban: igék illeszkedésének száma / tokenek száma.

1073 A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

1076 A forrás- és a célmondatban lévő írásjelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

1067 A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

DOI:10.15774/PPKE.ITK.2019.003

2015	**WordNet illeszkedés a célmondatban: igék illeszkedésének száma / tokenek száma.**
2029	**WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / határozószók száma.**
1038	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,05), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
2007	**WordNet illeszkedés F-mértéke: igék illeszkedésének száma / tokenek száma.**

A.4. táblázat OptTA 29 jegye

Az A.5. táblázatban található a gördülékenység értékekre (GA) betanított minőségbecslő modell optimalizált, 32 jegyből álló jegykészlete (OptGA), a relevancia sorrendjében. A vastagon szedett sorok jelzik az általam készített szemantikai jegyeket.

Azonosító	Leírás
1015	A célnyelvi szó előfordulásának száma a célnyelvi hipotézisben.
1060	Forrásnyelvi korpuszban lévő különböző trigramok aránya (minden kvartilisben).
1002	Tokenek száma a célmondatban.
1082	A forrás- és a célmondatban lévő csak a-z betűt tartalmazó tokenek aránya.
1091	Igék aránya a célmondatban.
2019	**WordNet illeszkedés a célmondatban: igék illeszkedésének száma / igék száma.**
1066	A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége.
2003	**Szótári illeszkedés F-mértéke.**
1036	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,01), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
1068	A forrás- és a célmondatban lévő pontosvesszők számának abszolút értékben vett különbsége.

DOI:10.15774/PPKE.ITK.2019.003

1072 A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége.

2020 WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / melléknevek száma.

2026 WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / főnevek száma.

1006 Tokenek átlagos hossza a forrásmondatban.

1010 Forrásmondat perplexitása.

1089 Igék aránya a forrásmondatban.

1044 Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}5$), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.

1073 A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

1054 Forrásnyelvi átlagos trigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1046 Forrásnyelvi átlagos unigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1093 Igék aránya a forrás- és a célmondatban.

2005 A forrás- és a célmondatban lévő NP-k számának abszolút értékben vett különbsége, a kifejezési címkék számával normalizálva.

2007 WordNet illeszkedés F-mértéke: igék illeszkedésének száma / tokenek száma.

2016 WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / tokenek száma.

1067 A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

1011 Forrásmondat perplexitása mondatvégi írásjel nélkül.

1052 Forrásnyelvi átlagos bigram gyakoriság a harmadik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

2001 Szótári illeszkedés a célmondatban.

DOI:10.15774/PPKE.ITK.2019.003

1034	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,5), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.
1042	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,2), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
2002	**Szótári illeszkedés a forrásmondatban.**
2015	**WordNet illeszkedés a célmondatban: igék illeszkedésének száma / tokenek száma.**

A.5. táblázat OptGA 32 jegye

Az A.6. táblázatban található a TA és a GA átlagának értékeire (TG) betanított minőségbecslő modell optimalizált, 26 jegyből álló jegykészlete (OptTG), a relevancia sorrendjében. A vastagon szedett sorok jelzik az általam készített szemantikai jegyeket.

Azonosító	Leírás
1015	A célnyelvi szó előfordulásának száma a célnyelvi hipotézisben.
1091	Igék aránya a célmondatban.
1089	Igék aránya a forrásmondatban.
1002	Tokenek száma a célmondatban.
1082	A forrás- és a célmondatban lévő csak a-z betűt tartalmazó tokenek aránya.
1066	A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége.
1044	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,5), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
1057	Forrásnyelvi átlagos trigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
2016	**WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / tokenek száma.**
1010	Forrásmondat perplexitása.

DOI:10.15774/PPKE.ITK.2019.003

1072	A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége.
2019	**WordNet illeszkedés a célmondatban: igék illeszkedésének száma / igék száma.**
1006	Tokenek átlagos hossza a forrásmondatban.
1068	A forrás- és a célmondatban lévő pontosvesszők számának abszolút értékben vett különbsége.
2005	A forrás- és a célmondatban lévő NP-k számának abszolút értékben vett különbsége, a kifejezési címkék számával normalizálva.
2001	**Szótári illeszkedés a célmondatban.**
1080	Tokenek száma a forrásmondatban, amelyek nem csak a-z betűt tartalmaznak.
2028	**WordNet illeszkedés a forrásmondatban: melléknevek illeszkedésének száma / melléknevek száma.**
1013	Célmondat perplexitása.
1052	Forrásnyelvi átlagos bigram gyakoriság a harmadik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
2022	**WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / tokenek száma.**
1073	A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1077	Számok aránya a forrásmondatban.
2006	**WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / tokenek száma.**
1067	A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1079	A forrás- és a célmondatban lévő számok számának abszolút értékben vett különbsége, a forrásmondat hosszával normalizálva.

A.6. táblázat OptTG 26 jegye

DOI:10.15774/PPKE.ITK.2019.003

Az A.7. táblázatban található a TA értékeiből készült osztályozási értékeire (CLTA) betanított minőségbecslő modell optimalizált, 21 jegyből álló jegykészlete (OptCLTA), a relevancia sorrendjében. A vastagon szedett sorok jelzik az általam készített szemantikai jegyeket.

Azonosító	Leírás
1068	A forrás- és a célmondatban lévő pontosvesszők számának abszolút értékben vett különbsége.
1064	A forrás- és a célmondatban lévő vesszők számának abszolút értékben vett különbsége.
1005	Tokenek számának abszolút értékben vett különbsége a forrás- és a célmondatban, a forrásmondat hosszával normalizálva.
1091	Igék aránya a célmondatban.
1092	Főnevek aránya a forrás- és a célmondatban.
1015	A célnyelvi szó előfordulásának száma a célnyelvi hipotézisben.
2001	**Szótári illeszkedés a célmondatban.**
1072	A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége.
1046	Forrásnyelvi átlagos unigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1077	Számok aránya a forrásmondatban.
1078	Számok aránya a célmondatban.
1055	Forrásnyelvi átlagos trigram gyakoriság a második kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1082	A forrás- és a célmondatban lévő csak a-z betűt tartalmazó tokenek aránya.
1066	A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége.
1093	Igék aránya a forrás- és a célmondatban.
1057	Forrásnyelvi átlagos trigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1081	Tokenek aránya a célmondatban, amelyek nem csak a-z betűt tartalmaznak.

DOI:10.15774/PPKE.ITK.2019.003

2019	**WordNet illeszkedés a célmondatban: igék illeszkedésének száma / igék száma.**
1067	A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1090	Főnevek aránya a célmondatban.
1010	Forrásmondat perplexitása.

A.7. táblázat OptCLTA 21 jegye

Az A.8. táblázatban található a GA értékeiből készült osztályozási értékeire (CLGA) betanított minőségbecslő modell optimalizált, 10 jegyből álló jegykészlete (OptCLGA), a relevancia sorrendjében. A vastagon szedett sorok jelzik az általam készített szemantikai jegyeket.

Azonosító	Leírás
1064	A forrás- és a célmondatban lévő vesszők számának abszolút értékben vett különbsége.
1076	A forrás- és a célmondatban lévő írásjelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
2002	**Szótári illeszkedés a forrásmondatban.**
1091	Igék aránya a célmondatban.
1072	A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége.
1047	Forrásnyelvi átlagos unigram gyakoriság a második kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1077	Számok aránya a forrásmondatban.
1011	Forrásmondat perplexitása mondatvégi írásjel nélkül.
1014	Célmondat perplexitása mondatvégi írásjel nélkül.
1054	Forrásnyelvi átlagos trigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

A.8. táblázat OptCLGA 10 jegye

DOI:10.15774/PPKE.ITK.2019.003

Az A.9. táblázatban található a TG értékeiből készült osztályozási értékeire (CLTG) betanított minőségbecslő modell optimalizált, 12 jegyből álló jegykészlete (OptCLTG), a relevancia sorrendjében. A vastagon szedett sorok jelzik az általam készített szemantikai jegyeket.

Azonosító	Leírás
1064	A forrás- és a célmondatban lévő vesszők számának abszolút értékben vett különbsége.
1091	Igék aránya a célmondatban.
1075	Írásjelek száma a célmondatban.
1093	Igék aránya a forrás- és a célmondatban.
1057	Forrásnyelvi átlagos trigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1072	A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége.
2010	**WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / főnevek száma.**
2025	**WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / tokenek száma.**
1066	A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége.
1014	Célmondat perplexitása mondatvégi írásjel nélkül.
1067	A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1079	A forrás- és a célmondatban lévő számok számának abszolút értékben vett különbsége, a forrásmondat hosszával normalizálva.

A.9. táblázat OptCLTG 12 jegye

DOI:10.15774/PPKE.ITK.2019.003

B. függelék

A kompozit rendszerhez felhasznált jegyek

A jelen függelékben találhatóak a kompozit rendszerhez használt minőségbecslő rendszerek tanításához és teszteléséhez felhasznált jegyek.

B.1. Felhasznált black-box jegyek

A B.1. táblázatban található az összes általam felhasznált 67 black-box jegy, amelyeket Lucia és társai [58] implementáltak.

Azonosító	Leírás
1001	Tokenek száma a forrásmondatban.
1002	Tokenek száma a célmondatban.
1003	Tokenek aránya a forrás- és a célmondatban.
1004	Tokenek száma a célmondatban / Tokenek száma a forrásmondatban.
1005	Tokenek számának abszolút értékben vett különbsége a forrás- és a célmondatban, a forrásmondat hosszával normalizálva.
1006	Tokenek átlagos hossza a forrásmondatban.
1007	Hibás zárójelek száma.
1008	Hibás idézőjelek száma.
1009	Forrásmondat nyelvmodell valószínűsége.
1010	Forrásmondat perplexitása.

DOI:10.15774/PPKE.ITK.2019.003

1011	Forrásmondat perplexitása mondatvégi írásjel nélkül.
1012	Célmondat nyelvmodell valószínűsége.
1013	Célmondat perplexitása.
1014	Célmondat perplexitása mondatvégi írásjel nélkül.
1015	A célnyelvi szó előfordulásának száma a célnyelvi hipotézisben.
1016	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,01).
1018	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,05).
1020	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,1).
1022	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,2).
1024	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,5).
1026	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,01), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.
1028	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,05), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.
1030	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,1), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.
1032	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,2), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.
1034	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség > 0,5), a forráskorpuszban lévő minden szó gyakoriságával súlyozva.

DOI:10.15774/PPKE.ITK.2019.003

1036	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}01$), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
1038	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}05$), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
1040	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}1$), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
1042	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}2$), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
1044	Fordítások átlagos száma / szavak száma a forrásmondatban (giza1 küszöb: valószínűség $> 0{,}5$), a forráskorpuszban lévő minden szó inverz gyakoriságával súlyozva.
1046	Forrásnyelvi átlagos unigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1047	Forrásnyelvi átlagos unigram gyakoriság a második kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1048	Forrásnyelvi átlagos unigram gyakoriság a harmadik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1049	Forrásnyelvi átlagos unigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1050	Forrásnyelvi átlagos bigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1051	Forrásnyelvi átlagos bigram gyakoriság a második kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1052	Forrásnyelvi átlagos bigram gyakoriság a harmadik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.
1053	Forrásnyelvi átlagos bigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

DOI:10.15774/PPKE.ITK.2019.003

1054 Forrásnyelvi átlagos trigram gyakoriság az első kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1055 Forrásnyelvi átlagos trigram gyakoriság a második kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1056 Forrásnyelvi átlagos trigram gyakoriság a harmadik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1057 Forrásnyelvi átlagos trigram gyakoriság a negyedik kvartilisben (kis gyakoriságú szavak), a forrásnyelvi korpuszban.

1058 Forrásnyelvi korpuszban lévő különböző unigramok aránya (minden kvartilisben).

1059 Forrásnyelvi korpuszban lévő különböző bigramok aránya (minden kvartilisben).

1060 Forrásnyelvi korpuszban lévő különböző trigramok aránya (minden kvartilisben).

1061 Átlagos szógyakoriság: forrásmondatban lévő minden type (unigram), ami a x-szer feltűnik a korpuszban (minden kvartilisben).

1062 A forrás- és a célmondatban lévő pontok számának abszolút értékben vett különbsége.

1063 A forrás- és a célmondatban lévő pontok számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

1064 A forrás- és a célmondatban lévő vesszők számának abszolút értékben vett különbsége.

1065 A forrás- és a célmondatban lévő vesszők számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

1066 A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége.

1067 A forrás- és a célmondatban lévő kettőspontok számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.

1068 A forrás- és a célmondatban lévő pontosvesszők számának abszolút értékben vett különbsége.

DOI:10.15774/PPKE.ITK.2019.003

1069	A forrás- és a célmondatban lévő pontosvesszők számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1070	A forrás- és a célmondatban lévő kérdőjelek számának abszolút értékben vett különbsége.
1071	A forrás- és a célmondatban lévő kérdőjelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1072	A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége.
1073	A forrás- és a célmondatban lévő felkiáltójelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1074	Írásjelek száma a forrásmondatban.
1075	Írásjelek száma a célmondatban.
1076	A forrás- és a célmondatban lévő írásjelek számának abszolút értékben vett különbsége, a célmondat hosszával normalizálva.
1077	Számok aránya a forrásmondatban.
1078	Számok aránya a célmondatban.
1079	A forrás- és a célmondatban lévő számok számának abszolút értékben vett különbsége, a forrásmondat hosszával normalizálva.
1080	Tokenek száma a forrásmondatban, amelyek nem csak a-z betűt tartalmaznak.
1081	Tokenek aránya a célmondatban, amelyek nem csak a-z betűt tartalmaznak.
1082	A forrás- és a célmondatban lévő csak a-z betűt tartalmazó tokenek aránya.

B.1. táblázat 67 black-box jegy a kompzit rendszerhez

A B.2. táblázatban található az összes általam készített 60 jegy, amelyeket felhasználtam az angol-magyar kompozit rendszer optimalizálásához.

Azonosító	**Leírás**
2001	Szótári illeszkedés a célmondatban.
2002	Szótári illeszkedés a forrásmondatban.
2003	Szótári illeszkedés F-mértéke.

DOI:10.15774/PPKE.ITK.2019.003

2006	WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / tokenek száma.
2007	WordNet illeszkedés F-mértéke: igék illeszkedésének száma / tokenek száma.
2008	WordNet illeszkedés F-mértéke: melléknevek illeszkedésének száma / tokenek száma.
2009	WordNet illeszkedés F-mértéke: határozószók illeszkedésének száma / tokenek száma.
2010	WordNet illeszkedés F-mértéke: főnevek illeszkedésének száma / főnevek száma.
2011	WordNet illeszkedés F-mértéke: igék illeszkedésének száma / igék száma.
2012	WordNet illeszkedés F-mértéke: melléknevek illeszkedésének száma / melléknevek száma.
2013	WordNet illeszkedés F-mértéke: határozószók illeszkedésének száma / határozószók száma.
2014	WordNet illeszkedés a célmondatban: főnevek illeszkedésének száma / tokenek száma.
2015	WordNet illeszkedés a célmondatban: igék illeszkedésének száma / tokenek száma.
2016	WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / tokenek száma.
2017	WordNet illeszkedés a célmondatban: határozószók illeszkedésének száma / tokenek száma.
2018	WordNet illeszkedés a célmondatban: főnevek illeszkedésének száma / főnevek száma.
2019	WordNet illeszkedés a célmondatban: igék illeszkedésének száma / igék száma.
2020	WordNet illeszkedés a célmondatban: melléknevek illeszkedésének száma / melléknevek száma.
2021	WordNet illeszkedés a célmondatban: határozószók illeszkedésének száma / határozószók száma.

DOI:10.15774/PPKE.ITK.2019.003

2022	WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / tokenek száma.
2023	WordNet illeszkedés a forrásmondatban: igék illeszkedésének száma / tokenek száma.
2024	WordNet illeszkedés a forrásmondatban: melléknevek illeszkedésének száma / tokenek száma.
2025	WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / tokenek száma.
2026	WordNet illeszkedés a forrásmondatban: főnevek illeszkedésének száma / főnevek száma.
2027	WordNet illeszkedés a forrásmondatban: igék illeszkedésének száma / igék száma.
2028	WordNet illeszkedés a forrásmondatban: melléknevek illeszkedésének száma / melléknevek száma.
2029	WordNet illeszkedés a forrásmondatban: határozószók illeszkedésének száma / határozószók száma.
10001	Igék aránya a mondatban.
10002	Főnevek aránya a mondatban.
10003	Melléknevek aránya a mondatban.
10004	Névmások aránya a mondatban.
10005	Határozószók aránya a mondatban.
10006	Kötőszók aránya a mondatban.
10007	Determinánsok aránya a mondatban.
10009	Számnevek aránya a mondatban.
10010	Írásjelek aránya a mondatban.
10011	Igekötők aránya a mondatban.
10012	Ismeretlen szavak aránya a mondatban.
10013	XML-címkék aránya a mondatban.
10101	Főnevek száma / igék száma.
10102	Főnevek száma / melléknevek száma.
10103	Főnevek száma / névelők száma.

DOI:10.15774/PPKE.ITK.2019.003

10104	Igék száma / igekötők száma.
10105	Mondatközi írásjelek száma / mondatvégi írásjelek száma.
10201	Tokenek száma a mondatban.
10202	Átlagos szóhossz a mondatban.
10203	Ékezetes karakterek száma a mondatban.
10204	Ékezetes szavak száma / tokenek száma a mondatban.
10301	A mondat szavainak n-gram valószínűsége.
10302	A mondat szavainak perplexitása (ismeretlen szavakkal együtt).
10303	A mondat szavainak perplexitása (ismeretlen szavak nélkül).
10304	A mondat szótöveinek n-gram valószínűsége.
10305	A mondat szótöveinek perplexitása (ismeretlen szavakkal együtt).
10306	A mondat szótöveinek perplexitása (ismeretlen szavak nélkül).
10307	A mondat elemzési címkéinek n-gram valószínűsége.
10308	A mondat elemzési címkéinek perplexitása (ismeretlen szavakkal együtt).
10309	A mondat elemzési címkéinek perplexitása (ismeretlen szavak nélkül).
10310	A mondat szófajcímkéinek n-gram valószínűsége.
10311	A mondat szófajcímkéinek perplexitása (ismeretlen szavakkal együtt).
10312	A mondat szófajcímkéinek perplexitása (ismeretlen szavak nélkül).

B.2. táblázat 60 jegy az angol-magyar kompzit rendszer optimalizálásához

DOI:10.15774/PPKE.ITK.2019.003

C. függelék

Az egynyelvű minőségbecsléshez felhasznált jegyek

C.1. Összes jegy

A C.1. táblázatban található az egynyelvű minőségbecslő modell tanításához felhasznált 36 jegy.

Azonosító	Leírás
10001	Igék aránya a mondatban.
10002	Főnevek aránya a mondatban.
10003	Melléknevek aránya a mondatban.
10004	Névmások aránya a mondatban.
10005	Határozószók aránya a mondatban.
10006	Kötőszók aránya a mondatban.
10007	Determinánsok aránya a mondatban.
10008	Indulatszók aránya a mondatban.
10009	Számnevek aránya a mondatban.
10010	Írásjelek aránya a mondatban.
10011	Igekötők aránya a mondatban.
10012	Ismeretlen szavak aránya a mondatban.
10101	Főnevek száma / igék száma.
10102	Főnevek száma / melléknevek száma.

DOI:10.15774/PPKE.ITK.2019.003

10103	Főnevek száma / névelők száma.
10104	Igék száma / igekötők száma.
10105	Mondatközi írásjelek száma / mondatvégi írásjelek száma.
10201	Tokenek száma a mondatban.
10202	Átlagos szóhossz a mondatban.
10203	Ékezetes karakterek száma a mondatban.
10204	Ékezetes szavak száma / tokenek száma a mondatban.
10301	A mondat szavainak n-gram valószínűsége.
10302	A mondat szavainak perplexitása (ismeretlen szavakkal együtt).
10303	A mondat szavainak perplexitása (ismeretlen szavak nélkül).
10304	A mondat szótöveinek n-gram valószínűsége.
10305	A mondat szótöveinek perplexitása (ismeretlen szavakkal együtt).
10306	A mondat szótöveinek perplexitása (ismeretlen szavak nélkül).
10307	A mondat elemzési címkéinek n-gram valószínűsége.
10308	A mondat elemzési címkéinek perplexitása (ismeretlen szavakkal együtt).
10309	A mondat elemzési címkéinek perplexitása (ismeretlen szavak nélkül).
10310	A mondat szófajcímkéinek n-gram valószínűsége.
10311	A mondat szófajcímkéinek perplexitása (ismeretlen szavakkal együtt).
10312	A mondat szófajcímkéinek perplexitása (ismeretlen szavak nélkül).
10401	A mondat 1-gram perplexitása (neurális nyelvmodell).
10402	A mondat 2-gram perplexitása (neurális nyelvmodell).
10403	A mondat 3-gram perplexitása (neurális nyelvmodell).

C.1. táblázat 36 jegy az egynyevű minőségbecslő modellhez

C.2. Optimalizált jegyek

A C.2. táblázatban található az egynyelvű minőségbecslő Likert-modell 15 jegyből álló, optimalizált jegykészlete, relevanciája szerint rendezve.

Azonosító	Jegy
10203	Ékezetes karakterek száma a mondatban.
10204	Ékezetes szavak száma / tokenek száma a mondatban.

DOI:10.15774/PPKE.ITK.2019.003

10302	A mondat szavainak perplexitása (ismeretlen szavakkal együtt).
10308	A mondat elemzési címkéinek perplexitása (ismeretlen szavakkal együtt).
10401	A mondat 1-gram perplexitása (neurális nyelvmodell).
10309	A mondat elemzési címkéinek perplexitása (ismeretlen szavak nélkül).
10012	Ismeretlen szavak aránya a mondatban.
10306	A mondat szótöveinek perplexitása (ismeretlen szavak nélkül).
10311	A mondat szófajcímkéinek perplexitása (ismeretlen szavakkal együtt).
10305	A mondat szótöveinek perplexitása (ismeretlen szavakkal együtt).
10008	Indulatszók aránya a mondatban.
10105	Mondatközi írásjelek száma / mondatvégi írásjelek száma.
10201	Tokenek száma a mondatban.
10101	Főnevek száma / igék száma.
10312	A mondat szófajcímkéinek perplexitása (ismeretlen szavak nélkül).

C.2. táblázat 15 jegyre optimalizált jegykészlet a Likert-modellhez

A C.3. táblázatban található az egynyelvű minőségbecslő osztályozási modell 28 jegyből álló, optimalizált jegykészlete, relevanciája szerint rendezve.

Azonosító	Jegy
10204	Ékezetes szavak száma / tokenek száma a mondatban.
10010	Írásjelek aránya a mondatban.
10203	Ékezetes karakterek száma a mondatban.
10004	Névmások aránya a mondatban.
10304	A mondat szótöveinek n-gram valószínűsége.
10001	Igék aránya a mondatban.
10103	Főnevek száma / névelők száma.
10101	Főnevek száma / igék száma.
10201	Tokenek száma a mondatban.
10307	A mondat elemzési címkéinek n-gram valószínűsége.
10006	Kötőszók aránya a mondatban.
10105	Mondatközi írásjelek száma / mondatvégi írásjelek száma.
10005	Határozószók aránya a mondatban.

DOI:10.15774/PPKE.ITK.2019.003

10310	A mondat szófajcímkéinek n-gram valószínűsége.
10302	A mondat szavainak perplexitása (ismeretlen szavakkal együtt).
10202	Átlagos szóhossz a mondatban.
10305	A mondat szótöveinek perplexitása (ismeretlen szavakkal együtt).
10003	Melléknevek aránya a mondatban.
10301	A mondat szavainak n-gram valószínűsége.
10311	A mondat szófajcímkéinek perplexitása (ismeretlen szavakkal együtt).
10402	A mondat 2-gram perplexitása (neurális nyelvmodell).
10009	Számnevek aránya a mondatban.
10104	Igék száma / igekötők száma.
10401	A mondat 1-gram perplexitása (neurális nyelvmodell).
10403	A mondat 3-gram perplexitása (neurális nyelvmodell).
10102	Főnevek száma / melléknevek száma.
10306	A mondat szótöveinek perplexitása (ismeretlen szavak nélkül).
10007	Determinánsok aránya a mondatban.

C.3. táblázat 28 jegyre optimalizált jegykészlet az egycímkés osztályozási modellhez

DOI:10.15774/PPKE.ITK.2019.003

D. függelék

A Fuzzy jegyekkel való kísérlet eredményei

A Fuzzy jegyeket tartalmazó egynyelvű jegyek és a HuQ korpusz segítségével felépítettem a minőségbecslő modelleimet, amelyekre optimalizálás is végeztem:

- LS modell: minőségbecslő modell Likert értékeket felhasználva.
- OS modell: minőségbecslő modell osztályzási értékeket felhasználva.
- OptLS modell: optimalizált LS modell.
- OptOS modell: optimalizált OS modell.

A D.1. táblázatban látható az 62 darab jegy, amelyekkel kísérleteztem.

Azonosító	**Leírás**
10001	Igék aránya a mondatban.
10002	Főnevek aránya a mondatban.
10003	Melléknevek aránya a mondatban.
10004	Névmások aránya a mondatban.
10005	Határozószók aránya a mondatban.
10006	Kötőszók aránya a mondatban.
10007	Determinánsok aránya a mondatban.
10008	Indulatszók aránya a mondatban.
10010	Írásjelek aránya a mondatban.

DOI:10.15774/PPKE.ITK.2019.003

10011	Igekötők aránya a mondatban.
10012	Ismeretlen szavak aránya a mondatban.
10013	XML címkék aránya a mondatban.
10101	Főnevek száma / igék száma.
10102	Főnevek száma / melléknevek száma.
10103	Főnevek száma / névelők száma.
10104	Igék száma / igekötők száma.
10201	Tokenek száma a mondatban.
10301	A mondat szavainak n-gram valószínűsége.
10302	A mondat szavainak perplexitása (ismeretlen szavakkal együtt).
10303	A mondat szavainak perplexitása (ismeretlen szavak nélkül).
10304	A mondat szavainak n-gram valószínűsége (mondatvégi írásjel nélkül).
10305	A mondat szavainak perplexitása (mondatvégi írásjel nélkül, ismeretlen szavakkal együtt).
10306	A mondat szavainak perplexitása (mondatvégi írásjel nélkül, ismeretlen szavak nélkül).
10307	A mondat szótöveinek n-gram valószínűsége.
10308	A mondat szótöveinek perplexitása (ismeretlen szavakkal együtt).
10309	A mondat szótöveinek perplexitása (ismeretlen szavak nélkül).
10310	A mondat elemzési címkéinek n-gram valószínűsége.
10311	A mondat elemzési címkéinek perplexitása (ismeretlen szavakkal együtt).
10312	A mondat elemzési címkéinek perplexitása (ismeretlen szavak nélkül).
10313	A mondat szófajcímkéinek n-gram valószínűsége.
10314	A mondat szófajcímkéinek perplexitása (ismeretlen szavakkal együtt).
10315	A mondat szófajcímkéinek perplexitása (ismeretlen szavak nélkül).
20001	Fuzzy egyezés szóbeágyazási modellel - Likert értéke.
20002	Fuzzy egyezés szóbeágyazási modellel - osztályozási értéke.
20003	Fuzzy egyezés szóbeágyazási modellel (csak szófajcímkék) - Likert értéke.
20004	Fuzzy egyezés szóbeágyazási modellel (csak szófajcímkék)- osztályozási értéke.
20005	Fuzzy egyezés LSA modellel - Likert értéke.

DOI:10.15774/PPKE.ITK.2019.003

20006	Fuzzy egyezés LSA modellel - osztályozási értéke.
20007	BLEU fuzzy egyezés - Likert értéke.
20008	BLEU fuzzy egyezés - osztályozási értéke.
20009	BLEU fuzzy egyezés szóbeágyazási modellel - Likert értéke.
20010	BLEU fuzzy egyezés szóbeágyazási modellel - osztályozási értéke.
20011	BLEU fuzzy egyezés LSA modellel - Likert értéke.
20012	BLEU fuzzy egyezés LSA modellel - osztályozási értéke.
20013	NIST fuzzy egyezés - Likert értéke.
20014	NIST fuzzy egyezés - osztályozási értéke.
20015	NIST fuzzy egyezés szóbeágyazási modellel - Likert értéke.
20016	NIST fuzzy egyezés szóbeágyazási modellel - osztályozási értéke.
20017	NIST fuzzy egyezés LSA modellel - Likert értéke.
20018	NIST fuzzy egyezés LSA modellel - osztályozási értéke.
20019	TER fuzzy egyezés - Likert értéke.
20020	TER fuzzy egyezés - osztályozási értéke.
20021	TER fuzzy egyezés szóbeágyazási modellel - Likert értéke.
20022	TER fuzzy egyezés szóbeágyazási modellel - osztályozási értéke.
20023	TER fuzzy egyezés LSA modellel - Likert értéke.
20024	TER fuzzy egyezés LSA modellel - osztályozási értéke.
20025	Levenstein fuzzy egyezés - Likert értéke.
20026	Levenstein fuzzy egyezés - osztályozási értéke.
20027	Levenstein fuzzy egyezés szóbeágyazási modellel - Likert értéke.
20028	Levenstein fuzzy egyezés szóbeágyazási modellel - osztályozási értéke.
20029	Levenstein fuzzy egyezés LSA modellel - Likert értéke.
20030	Levenstein fuzzy egyezés LSA modellel - osztályozási értéke.

D.1. táblázat 62 jegy az egynyevű minőségbecslő modellhez

A D.2. táblázatban és a D.3. táblázatban látható a modellek kiértékelése.

DOI:10.15774/PPKE.ITK.2019.003

	Korreláció ↑	MAE ↓	RMSE ↓
LS modell - 62 jegy	0,5936	0,6857	0,8961
OptLS modell - 13 jegy	0,6278	0,6783	0,8758

D.2. táblázat LS modell és OptLS modell kiértékelése

	CCI ↑	MAE ↓	RMSE ↓
OS modell - 62 jegy	70,7%	0,2465	0,3590
OptOS modell - 8 jegy	71,7%	0,2544	0,3539

D.3. táblázat OS modell és OptOS modell kiértékelése

A D.4. táblázatban és a D.5. táblázatban láthatóak az optimalizált jegyhalmazok, relevancia szerint sorbarendezve.

Azonosító	Leírás
10313	A mondat szófajcímkéinek n-gram valószínűsége.
10006	Kötőszók aránya a mondatban.
20001	Fuzzy egyezés szóbeágyazási modellel - Likert értéke.
10307	A mondat szótöveinek n-gram valószínűsége.
10002	Főnevek aránya a mondatban.
20016	NIST fuzzy egyezés szóbeágyazási modellel - osztályozási értéke.
10314	A mondat szófajcímkéinek perplexitása (ismeretlen szavakkal együtt).
10003	Melléknevek aránya a mondatban.
10010	Írásjelek aránya a mondatban.
10104	Igék száma / igekötők száma.
10011	Igekötők aránya a mondatban.
10012	Ismeretlen szavak aránya a mondatban.
20002	Fuzzy egyezés szóbeágyazási modellel - osztályozási értéke.

D.4. táblázat 13 jegyből álló optimalizált jegykészlet Likert modellhez

DOI:10.15774/PPKE.ITK.2019.003

Azonosító	Leírás
10301	A mondat szavainak n-gram valószínűsége.
10302	A mondat szavainak perplexitása (ismeretlen szavakkal együtt).
10006	Kötőszók aránya a mondatban.
20022	TER fuzzy egyezés szóbeágyazási modellel - osztályozási értéke.
20027	Levenstein fuzzy egyezés szóbeágyazási modellel - Likert értéke.
10303	A mondat szavainak perplexitása (ismeretlen szavak nélkül).
10308	A mondat szótöveinek perplexitása (ismeretlen szavakkal együtt).
10010	Írásjelek aránya a mondatban.

D.5. táblázat 8 jegyből álló optimalizált jegykészlet az osztályzási modellhez

Az eredményekből azt lehet leszűrni, hogy a szóbeágyazási modell jobban teljesít az LSA modellhez képest, hiszen az optimalizált halmazokba egy LSA jegy sem került bele. Továbbá az is látható, hogy a Fuzzy egyezés modellek is előkelő helyezéseket értek el. Ez pedig azt jelenti, hogy relevánsak az eredményre nézve.