Szövegszinkronizációs módszerek, hibrid bekezdés- és mondatszinkronizációs megoldás

Pohl Gábor

Pázmány Péter Katolikus Egyetem Információs Technológiai Kar
pohl@morphologic.hu

Kulcsszavak: szövegszinkronizáció (text alignment), mondatszinkronizáció (sentence alignment), bekezdésszinkronizáció (paragraph alignment), statiszikai módszerekkel szűrt horgonyok, hibrid szinkronizációs megoldás.

Kivonat: A cikkben bemutatjuk a szövegszinkronizáció általánosított definícióját; a szövegegységek hosszán alapuló szinkronizációt; a horgonyok használatának és statisztikai szűrésének lehetőségét; majd olyan, a két stratégiát ötvöző hibrid szinkoronizációs megoldást ismertetünk, amely beszúrásokat és elhagyásokat tartalmazó szövegpárok szinkronizálására alkalmasabb az eddigi módszereknél.

1 Szövegszinkronizáció

Szövegszinkronizáción (text alignment) két- vagy többnyelvű szövegekben az egymás fordításának tekinthető szövegegységek meghatározását értjük. Fordítómemória nélkül készített fordítások fordítómemóriákba töltéséhez, fordítások terminológiai konzisztenciájának gépi ellenőrzéséhez, nyelvészeti kutatások alapjául szolgáló párhuzamos korpuszok építéséhez legalább mondatszintű szinkronizációra van szükség. Az egymás fordításának tekinthető mondatok gépi meghatározása azonban nehéz feladat, mivel a fordítók a mondatok határait a fordítás során megváltoztathatják, (véletlenül) elhagyhatnak, illetve beszúrhatnak mondatokat, felcserélhetik a mondatok sorrendjét.

1.1 Általánosított definíció

A szövegszinkronizáció definícióját korábbi szerzők saját munkáikban különbözőképp határozták meg, hiszen az általuk megoldani kívánt szinkronizációs feladatok, is különbözők voltak. A következőkben az általunk használt általánosított definíciót [1] mutatjuk be.

Szövegszinkronizáción párhuzamos szövegek szinkronizációs egységeinek olyan egy-egyértelmű egymáshoz rendelését értjük (szinkronizációsegység-párokat határozunk meg), amely során egy adott forrásszövegbeli szinkronizációs egységhez egy

fordításbeli szinkronizációs egységét csak akkor rendelünk hozzá, ha az a teljes szövegeket tekintve a forrásszövegbeli szinkronizációs egység fordításának tekinthető.

Szinkronizációs egységen általánosságban a szöveg pontosan meghatározott típusú egységeinek – nem feltétlenül egymáshoz tapadó egységekből képzett – halmazát érjük, amely nem tartalmaz más szinkronizációs egységben szerepelő szövegbeli egységet. Ez a definíció tehát megengedi, hogy például két egymástól távol eső mondat (konkrét szövegbeli előfordulás) alkosson egy szinkronizációs egységet, de a szinkronizációs egységek között nem lehet átfedés, azaz az adott mondatoknak ezek az előfordulásai más szinkronizációs egységhez nem tartozhatnak.

A szövegszinkronizációs módszerek feladata a szinkronizációs egységek és a köztük levő kapcsolatok meghatározása. Cél, hogy az egyes szinkronizációs egységek minél kevesebb szövegbeli egységből álljanak, és csak akkor tartalmazzanak több szövegbeli egységet, ha ez mindenképp szükséges (például két forrásszövegbeli mondatot egy fordításbeli mondattá vont össze a fordító).

1.2 A szinkronizáció típusai

Attól függően, hogy a szinkronizációs egységek milyen szövegegységekből állnak, megkülönböztetünk bekezdésszintű, mondatszintű, kifejezésszintű illetve szószintű szinkronizációt. A szövegek lefedettsége alapján megkülönböztetünk teljes és részleges szinkronizációt. A szinkronizációt teljesnek tekintjük, ha a forrásszöveg és a fordítás valamennyi egymáshoz rendelhető szinkronizációs egységét meghatároztuk; részleges szinkronizáció esetén a szinkronizációsegység-párok nem fedik le a forrásszöveg és fordítás valamennyi lefedhető szövegegységét.

A szinkronizációs módszerek ezen kívül különbözhetnek az alkalmazott technológiákban, pontosságban lefedettségben és robusztusságban. A következőkben két alapvető stratégiát mutatunk be, a szövegegységek hosszán alapuló módszert, amely teljes szinkronizációt tesz lehetővé, illetve a horgonyokat használó részleges szinkronizációt megcélzó módszert.

2. Szövegegységek hosszán alapuló módszer

A legtöbb mondat- és bekezdésszinkronizáló módszer a szinkronizálandó mondatok vagy bekezdések valamilyen mérték szerinti hosszán, illetve a Gale és Church által publikált módszeren [2][3] alapul.

A szinkronizálásnál az egyes szövegegységek hosszát mérhetjük karakterekben, szavakban vagy például a jelentés átvitele szempontjából legfontosabb szófajú szavak számát meghatározva. A betűírást használó nyelvek esetében a karakterszám a másik két lehetőségnél pontosabb összehasonlítást tesz lehetővé [1][2]. A jelentés átvitele szempontjából fontos szófajú szavak számát összehasonlító módszerekre a nem betűírást használó nyelveknél van szükség.

Gale és Church a szövegegységek hosszainak valószínűségi modellen alapuló összehasonlítását választotta [2][3]. Az összehasonlítás azon a feltételezésen alapul, hogy amennyiben L1 és L2 nyelvű szövegek egymás fordításai, és az L1 nyelvű szöveg minden egyes karaktere véletlen számú L2-beli karakter megjelenését okozza,

akkor ezek az egyes karakterekhez rendelt valószínűségi változók függetlenek és mind azonos normál eloszlással jellemezhetők.

Mondat- illetve bekezdésszinkronizációs módszerek esetében meg kell határozni, hogy milyen szinkronizációsegység-pár típusokat kezelnek a módszerek. Legegyszerűbb esetben a fordító egy forrásnyelvi mondatot egy mondatban fordít le, ilyenkor a párosított szinkronizációs egységek közül a forrásszöveghez és a fordításhoz tartozó is egy-egy mondatot tartalmaz (1-1 megfeleltetés). Emellett célszerű kezelni a 0-1 (beszúrás), 1-0 (elhagyás) 1-2 (részekre bontás), 2-1 (összevonás) és a 2-2 (mondathatár eltolás) típusú megfeleltetéseket.

Gale és Church a mondatok szinkronizálására egy dinamikus programozásra épülő algoritmust javasolt. A dinamikus programozás olyan globális optimumkereső eljárás, amely a feladat optimális megoldását alkalmas részfeladatok megoldásával éri el. Mondatszinkronizáció esetében az egyes szinkronizációsegység-összerendelésekhez költségeket rendelünk, majd a költségek összegét akarjuk minimalizálni. A költségek Gale és Church módszere esetében az egymáshoz rendelt szövegrészek hosszaránya alpján számított költségből és az alkalmazott szinkronizációs kategóriához (összerendelés típushoz) rendelt költségből állnak össze.

A dinamikus programozás előnye, hogy biztosan megtalálja a legkisebb költségű teljes szinkronizációt, és hogy nem használ véletlent, így bármikor alkalmazzuk ugyanarra az eredményre jutunk. A módszer hátránya a viszonylag nagy számítási- és memóriaigénye: M szövegegységnek N szövegegységgel történő szinkronizációjakor (M+1)×(N+1)-es táblázatot kell memóriában tartani és kitölteni. A táblázatnak azonban elég csak az átló körüli elemeit kiszámítani [1] így lényegesen gyorsabb változata készíthető az algoritmusnak, amely nem globálisan optimális megoldást keres már, hiszen csak a várhatóan legjobb megoldásváltozatok közül választja ki a legjobbat.

Gale és Church módszere az elhagyott és beszúrt szövegegységek kezelésekor nem bizonyult elég pontosnak, ami nem csoda, hiszen a szövegeket pusztán különböző hosszúságú szövegegységek sorozataként tekinti: a szövegegységek hosszán kívül más információt nem használ fel. Egyszerű (sok 1-1 megfeleltetést tartalmazó) szövegpárok esetében a módszerrel 95% feletti pontosságot értek el, azonban nehezebb szövegek esetében a módszer nem működött ilyen jól [5][6].

3. Horgonykeresés

Szövegpárok részleges szinkronizálása érdekében a szövegek pontjai között egy-egyértelmű megfeleltetést teremtve horgonynak (anchor) nevezett kapcsolatok kereshetők. Horgonyt alkothat bármilyen szövegegység-pár, ha a két szövegben egymás megfelelőinek tekinthetők.

3.1 A lehetséges horgonyok kiválasztása

Horgonyként, azonos karakterkészlettel írt nyelvek esetében választhatók a két szövegben előforduló azonos alakú szavak (homograph). Simard, Foster és Isabelle

annak érdekében, hogy minél több horgonyt találjanak, a hasonló alakú szavakat (cognate) is alkalmasnak találták a két szöveg közti kapcsolatok felvételére [5].

A különböző horgonytípusok közti választás során a legfontosabb szempontként a gyakoriságukat, illetve feltételezhető megbízhatóságukat érdemes megvizsgálni. A megbízhatóság érdekében a hibásan felvett párokat valamilyen módszerrel érdemes szűrni. A horgonykeresés az agglutináló és flektáló nyelveknél nehézségekbe ütközhet, hiszen a szóalakok a toldalékolásnak, illetve hajlításnak megfelelően változnak.

3.2 Hibás horgonyok szűrése statisztikai módszerekkel

A hibásan felvett horgonyok szűrésére eddig a legmegbízhatóbbnak tekinthető módszert Ribeiro, Lopes és Mexia publikálta [4]. A korábban ismertetett heurisztikus módszerekkel ellentétben Ribeiro és társai két statisztikai szűrőt definiáltak: mindkettőt a horgonyjelöltek szövegbeli pozíciói alapján kiszámítható lineáris regressziós sáv alkalmazásával. Első lépésben a lineáris regressziós sáv körül egy adaptív hisztogram alapú szűrővel meghatározott tartományon kívül eső pontokat vetették el, majd a regressziós sáv konfidenciasávján kívüli pontokkal tették ugyanezt.

4. Hibrid szinkronizációs megoldás

A szövegegységek hosszán alapuló módszer előnye, hogy elég robusztus, nem függ attól, hogy találhatók-e megfelelően pontos horgonyok a szövegpárban, ugyanakkor a beszúrásokat, elhagyásokat rosszul kezeli a módszer. Horgonyok használatával pontos, de csak részleges szinkronizáció érhető el. A horgonyok által elért részleges szinkronizáció sajnos nem alkalmas a szöveg kisebb szinkronizálandó szegmensekre bontására [1], így a két módszert csak egyetlen szinkronizáló algoritmusba integrálva lehet ötvözni. A következőkben az általunk választott hibrid, horgonyokon és szövegegységhosszakon alapuló megoldást mutatjuk be.

4. 1 Horgonyok keresése és szűrése

Legmegbízhatóbb horgonynak a szövegekben azonos számban előforduló, nagybetűket vagy számjegyeket tartalmazó szavakat választottuk. A számok esetében célszerűnek találtuk a tizedespontok, tizedesvesszők, és az esetlegesen a számjegyek között előforduló szóközök törlését az összehasonlítás előtt, így (a latin betűs nyelvek esetében) nyelvfüggetlen számformátumot hozva létre. A számjegyek között megengedtünk egyéb karaktereket is, így a termékkódokat, telefonszámokat és egyéb számokat tartalmazó szavak is horgonypontokká válhattak.

A magyar toldalékolás következtében az egyes szavak szóalakja változhat, ez ellen kétféleképp lehet tenni: morfológia alkalmazásával vagy Simardhoz és társaihoz hasonlóan az azonosan kezdődő szavak keresésével. Az ismeretlen szavak kezelésére is alkalmas morfológia alkalmazásával pontosabb megoldás készíthető, így a szavak morfológiai rendszerekkel történő tövesítése mellett döntöttünk.

A horgonyok szűrésére Ribeiro, Lopes és Mexia statisztikai alapú megoldásait [4] választottuk, amelyeket az előzőekben már röviden ismertettünk.

4.2 Hibrid algoritmus

A Gale és Church által kidolgozott szöveghosszakon alapuló algoritmust [2][3] vettük alapul, azonban a dinamikus programozás során nem csak a szövegegységek hosszának arányán és a választott szinkronizációs kategórián alapuló szinkronizációs költséget használjuk fel, hanem – előre meghatározott súllyal figyelembe véve – a horgonyok alapján számított (a költségekkel ellentétes előjelű) hasznot is.

A horgonyok alapján meghatározott haszon kiszámítására egy olyan egyszerű, heurisztikus módszert alkalmazunk, amely a fordítás során történő beszúrások és elhagyások felismerését elősegíti. A heurisztika alkalmazása elkerülhetetlen, mivel nincsen tudományos alapokon nyugvó elmélet, amely a feladat megoldása során alkalmazható lenne. A szövegegység-hosszakon alapuló módszerrel való kombinálhatósághoz egy valószínűségi alapokon nyugvó horgonyelőfordulás modell lenne szükséges, ilyet azonban már csak azért sem alkothatunk, mert a horgonyok várható eloszlását nem ismerjük. (A horgonyok eloszlása nem tekinthető egyenletesnek, a szövegekben csomósodást mutatva néhol egyszerre több horgony szerepel egy helyen; máskor ritkábbak, néhol pedig teljesen hiányoznak a szövegből a horgonynak tekinthető párok. Annak ismeretében, hogy a szövegek egyes részei más és más szerepet töltenek be, ez a jelenség nem tekinthető különösnek, a szövegekről alkotott képünkkel egybevág.)

A heurisztikus megoldás szükségszerűségének rövid indoklása után kezdjük rögtön a legfontosabbal, a horgonyok alapján számított haszon kiszámításának javasolt módjával. A hasznot a következő heurisztikus képlettel számítjuk:

$$haszon = \frac{\dfrac{\textit{a résztvevő szövegegységek közös horgonyainak száma}}{\textit{a résztvevő szövegegységek összes horgonyának száma}}}{\textit{résztvevő szövegegységek száma}} \qquad (1)$$

Az (1) képletet akkor alkalmazzuk, ha vannak horgonyaink (ha nincsenek, akkor értelemszerűen csak a szövegegységhosszak alapján lehetséges a szinkronizációs költség meghatározása). A képlet fő törtjének számlálójában levő törtkifejezés azt részesíti előnyben, ha az adott szinkronizációs kategória által meghatározott szövegegységekhez tartozó horgonyok nagy része a szinkronizációs kategória által meghatározott szövegegységeken belüli szövegpontokat köt össze (ezáltal szükség esetén az összevonásokat preferálva, ha ezt közös horgonyok indokolják). A nevező ezzel szemben a felesleges összevonásokat próbálja elkerülni, azáltal, hogy kisebb hasznot rendel a több szövegegységet összevonó szinkronizációs kategóriákhoz.

A fenti heurisztika nem preferálja, ha egy szinkronizációsegység-pár több horgonnyal is össze van kapcsolva: azt részesíti csak előnyben, ha a horgonyok közül minél több a szinkronizációsegység-páron belüli pontokat köt össze. Bonyolultabbá tenné a módszert, ha az összekötő horgonyok abszolút számát is fel akarnánk használni, hiszen ekkor a résztvevő szövegegységek hosszát is számításba kellene vennünk, mivel hosszabb szövegegységen belül nagyobb valószínűséggel találunk (több) horgonyt. A horgonyok nem egyenletes eloszlása miatt azonban nem tartottuk

szerencsésnek a szöveghosszal való összevetést. Bár a módszer feltehetően működne, az első kísérletekhez megfelelőbbnek találtuk egy egyszerű és átlátható heurisztika választását (amit az eredmények, – úgy tűnik –, utólag igazolnak is).

Az (1) képlet törtjének nevezőjébe akkor is beszámítjuk a résztvevő szövegegységet, ha az adott szövegegység nem is tartalmaz horgonypontot (csak egy vagy több másik, az adott szinkronizációsegység-pár részét képező szövegegység). Ez az eljárás megkérdőjelezhető, későbbi kísérletekkel kell majd eldönteni, hogy érdemes-e így eljárni. A szövegegység beszámításának előnye, hogy a beszúrt (vagy elhagyott) szövegegységek esetében nem támogatja a beszúrt (vagy elhagyott) szövegegység hozzávételét egy másik szövegegységpárhoz. Hátránya lehet, hogy akkor sem támogatja a 2-1 vagy 1-2 típusú összerendeléseket, ha azok szükségesek. Megoldás lehet, ha a horgonyokon alapuló heurisztikusan számított haszon számításba vételét meghatározó súlyt úgy határozzuk meg, hogy amikor a szövegegység hosszakon alapuló módszer az adott összerendelést nagymértékben támogatja, akkor az összerendelés létrejöhessen.

4.3 Eredmények

Az előzőekben ismertetett hibrid módszert olyan angol illetve magyar nyelvű informatikai témájú szövegeken teszteltük, amelyek a csak szövegegységhosszakon alapuló módszerrel a beszúrások és elhagyások miatt gyakorlatilag szinkronizálhatatlanok voltak. A problémás helyek közelében a hibrid módszer a horgonyoknak köszönhetően többnyire helyesen határozta meg a szinkronizációs egységeket. Horgonyok hiányában természetesen a csak szöveghosszakon alapuló módszerrel azonos eredményt ért el a hibrid megoldás is, ezért a későbbiekben a megbízható horgonyok számának növelésével a már most is megfelelő eredmények tovább javíthatók majd.

Referenciák

1. Pohl Gábor: Fordítások terminológiai konzisztenciájának vizsgálata (diplomaterv feladat). Budapesti Műszaki és Gazdaságtudományi Egyetem (2003)
2. Gale, William A.; Kenneth W. Church: A Program for Aligning Sentences in Bilingual Corpora. In: 29th Annual Meeting of the Association for ational Linguistics (1991)
3. Gale, William A.; Kenneth W. Church: A Program for Aligning Sentences in Bilingual Corpora. Computational Linguistics, Volume 19, Number 1, March 1993, Special Issue on Using Large Corpora: I.
4. Ribeiro, António; Gabriel Lopes; João Mexia: Using Confidence Bands for Parallel Texts Alignment. In: Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (2000).
5. Simard, Michael; Foster, George; Isabelle, Pierre: Using Cognates to Align Sentences in Bilingual Corpora. In: Proceedings of TMI-92, Montréal, Canada. (1992) pp. 67-81
6. Langlais, Philippe; Michel Simard; Jean Véronis: Methods and Practical Issues in Evaluating Alignment Techniques. In: 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (1998)