Interlinking Korean Resources on the Web

Soon Gill Hong[1], Saemi Jang[1], Young Ho Chung[1],
Mun Yong Yi[1], and Key-Sun Choi[2]

[1] Department of Knowledge Service Engineering, KAIST, Republic of Korea
{hsoongil,sammyjang,nowespy}@gmail.com,
munyi@kaist.ac.kr
[2] Department of Computer Science, KAIST, Republic of Korea
kschoi@kaist.ac.kr

Abstract. LOD (Linked Open Data) is an international endeavor to interlink structured data on the Web and create the Web of Data on a global level. In this paper, we report about our experience of applying existing LOD frameworks, most of which are designed to run only in European language environments, to Korean resources to build linked data. Through the localization of Silk, we identified localized similarity measures as essential for interlinking Korean resources. Specifically, we built new algorithms to measure distance between Korean strings and to measure distance between transliterated Korean strings. A series of empirical tests have found that the new measures substantially improve the performance of Silk with high precision for matching Korean strings and with high recall for matching transliterated Korean strings. We expect the localization issues described in this paper to be applicable to many non-Western countries.

Keywords: LOD, Silk, Distance measure, Localization, Transliteration.

1 Introduction

One serious drawback of the current Web is its limited support for sharing and interlinking online resources at the data level. Linked Open Data (LOD) is an international endeavor to overcome this limitation of the current Web and create the Web of Data on a global level. Since Web 2.0 emerged, a great amount of data has been released to the LOD cloud in the structured-data format so that computers can understand and interlink them. Many tools and frameworks have been developed to support the transition and successfully deployed in a wide number of areas. However, as the proportion of non-Western data is comparatively small, a couple of projects have been only recently initiated to extend the boundary of LOD over non-Western language resources.

The goal of this research is to report about our experience of applying existing linked open data frameworks, most of which are designed to run only in European language environments, to Korean resources to build linked data. In addition to inherent multi-byte issues related to non-European languages [1] [2], we identified key localization issues that should be taken into account when building links among

H. Takeda et al. (Eds.): JIST 2012, LNCS 7774, pp. 382–387, 2013.

multilingual resources. In particular, we have developed two new Korean similarity metrics and implemented those metrics into Silk, a tool specifically designed for linking LOD resources [3], and compared the performance of the new metrics to Levenshtein Distance. Through a series of empirical tests, we have confirmed that the new metrics offer several advantages over the existing metrics.

2 Related Work

The Korean alphabet system, Hangul, is the native alphabet of the Korean language, consisting of fourteen consonant letters (i.e., 'ㄱ', 'ㄴ', 'ㄷ', 'ㄹ', 'ㅁ', 'ㅂ', 'ㅅ', 'ㅇ', 'ㅈ', 'ㅊ', 'ㅋ', 'ㅌ', 'ㅍ', and 'ㅎ') and ten vowel letters (i.e., 'ㅏ', 'ㅑ', 'ㅓ', 'ㅕ', 'ㅗ', 'ㅛ', 'ㅜ', 'ㅠ', 'ㅡ', and 'ㅣ'). Two consonant letters can be combined to create consonant digraphs (i.e., 'ㄲ'or 'ㄺ'), and two or three vowel letters can be combined to create vowel digraphs or trigraphs (i.e., 'ㅘ' or 'ㅞ'). Syllables are composed by combining one consonant (letter or digraph), one vowel (letter, digraph, or trigraph), and one optional consonant (letter or digraph). The current Unicode system contains 11,172 syllables, which can cover all of the modern Korean words.

Levenshtein Distance, which is also known as 'edit distance', is a popular metric to measure distance between two Latin Alphabet strings and highly recommended to use in Silk. Levenshtein distance is defined as the minimum number of insertion, deletion, or substitution operation of a single character needed to transform one string into the other. Soundex is a widely used phonetic similarity measure for English to score the distance between strings based on not letters but sound.

In the Unicode system, the unit of comparison of English alphabet is one letter because each English letter is assigned to one code point. In Korean, however, a combination of 2 or 3 letters (diagraphs or trigraphs, hereafter letter), representing a syllable in Korean, is assigned to one code point. Thus, even though Levenshtein Distance says that the distance is one, it could mean one, two, or three different letters in Korean. Thus, research has been conducted to develop localized similarity measures for Korean strings. For example, KorED computes distance between two strings by calculating the number of necessary syllable and phoneme operations of insertion, deletion, or substitution to make them identical. GrpSim and OneDSim2 have similar approaches except assigning different weights based on the sound or the location of the phonemes [4].

Transliteration converts letters from one writing system into another and doesn't concern representing original phonemes. For example, one of the Korean popular food "칼국수" (knife-cut Korean noodles in translation) can be transliterated as "Kalguksu" in English. To the best of our knowledge, there are no such metrics to measure similarity between transliterated Korean strings. Instead, there are several studies on transliteration of English to Korean and back-transliteration of Korean to English [5] [6] and there is a study on a similarity measure for Korean transliteration of foreign words using algorithms similar to Soundex [7].

3 Linking Korean Resources with New Phonemic and Phonetic Measures

3.1 New Korean Similarity Measure by Phonemic Distribution

Our approach is based on the idea that the more the phonemes are distributed across the syllables, the less the possibility there is that the strings have the same meaning. Using this new algorithm, we can control the range of the target string more precisely, especially for those string pairs that have only one or two phonemes different from only one syllable. For example, if we specify a threshold of 2 (phonemes), then search for "호랑이" ("tiger" in English and "horangi" in English transliteration) would retrieve its dialect "호래이" ("tiger" in English and "horaei" in English transliteration) as a candidate correctly but not retrieve "오라이" ("OK" in English and "orai" in English transliteration).

This is how our proposed algorithm works; it calculates the number of different syllables. Then it chooses one random syllable with the least number of different phonemes and then regards the number of different phonemes in that syllable as 'α'. The number of different syllables minus one is regarded as 'β'. We can get the final phoneme distance by the formula shown in Eq. (1).

$$\text{Korean Phoneme Distance} = \begin{cases} (\text{sD} - 1) * 3 + \min[\text{pD}_\text{n}] \; if \; sD > 0 \\ 0 \; else \end{cases} \tag{1}$$

In Eq. (1), sD means syllable distance, and pD_n is a list of phoneme distances of syllables. The multiplier 3 is a weighting factor for a syllable because the most common number of phonemes in one Korean syllable is 3.

We tested the proposed algorithm using all of the Korean strings that appear in the English DBpedia. Assuming that the number of relevant records by Levenshtein Distance is the actual number of relevant records, we obtained the performance results summarized in Figure 1. With threshold 1, the recall score of Korean Phoneme Distance (KoPhoDist in the figure, hereafter Phoneme Distance) was almost the same as that of Levenshtein Distance (99.24% vs. 100%), but the precision score was substantially higher than that of Levenshtein Distance (81.23% vs. 21.40%).

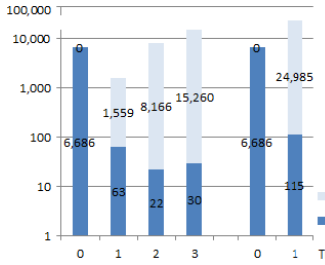
	Precision(%)	Recall(%)	F-Score	Retrieved	Relevant	Ret. & Rel.
KoPhoDist[1]	81.23	99.24	0.8934	8,308	6,801	6,749
Levenshtein[1]	21.40	100.00	0.3525	31,786		6,801

[1] threshold=1

Fig. 1. Performance comparison for DBpedia data

From the results, we can find that, with the Phoneme Distance measure the F-score is 0.8934, and with the Levenshtein Distance measure the F-score is 0.3525, showing that the newly developed measure is about two-and-a-half times more effective in finding correct links. This test revealed that Levenshtein measure retrieved a few more correct links at the cost of a formidably large number of incorrect links.

We can further see the benefits of Phoneme Distance by incrementally changing the search scope as shown in Figure 2. Phoneme and Levenshtein Distance produce the same results when the threshold is set to 0. When the threshold is changed to 1, Levenshtein Distance retrieves an extra of 25,100, of which 115 records (0.46%) are relevant and 24,985 records (99.54%) are irrelevant. The figure shows that this formidable amount of the irrelevant records can be effectively controlled by using Phoneme Distance. Phoneme Distance retrieves an extra of 1,662 (63 relevant and 1,559 irrelevant records) with the threshold of 1, an extra of 8,188 (22 relevant and 8,166 irrelevant records) with the threshold of 2, and an extra of 15,290 (30 relevant and 15,260 irrelevant records) with the threshold of 3. It should be noted that the portion of the relevant records is decreasing (3.79% to 0.27% to 0.20%) as the threshold value is incrementally increasing (1 to 2 to 3), clearly indicating that the incentive to include additional range of search is diminished rapidly.



Distance Measure	Phoneme					Levenstein		
Threshold	0	1	2	3	total	0	1	total
Retrieved & Not Relevant	0	1,559	8,166	15,260	24,985	0	24,985	24,985
Retrieved & Relevant	6,686	63	22	30	6,801	6,686	115	6,801
Total	6,686	1,622	8,188	15,290	31,786	6,686	25,100	31,786

Fig. 2. Comparative Search Results - Phoneme Distance vs. Levenshtein Distance

3.2 Korean Transliteration Similarity Measure by Phonetic Features

The best approach to measure distance between transliterated strings would be employing back transliteration of those transliterated strings into the original language and then applying string similarity metrics. This approach, however, is impractical for Korean because a transliterated Korean string could be transliterated back into several possible Korean strings. Due to this difficulty, we decided to take a simpler but more practical approach for measuring similarity between transliterated Korean strings. Our new algorithm replaces 'k' with 'g, 't' with 'd', 'p' with 'b', 'r' with 'l', and then applies Levenshtein Distance on the converted strings. The biggest difference between Soundex and our scheme of Transliterated Korean Distance (hereafter Transliterated Distance) is that we don't eliminate vowels, the other consonants, and any duplicates. And we don't limit the number of letters for comparison.

We tested two cases with each different threshold for Revised Romanization Transliteration (the current standard in Korea). As summarized in Figure 3, with threshold 0 the precision score of Transliterated Distance (KoTrlitDist in the figure) was almost the same as that of Levenshtein Distance (99.86% vs. 99.98%), but the recall score was higher than that of Levenshtein Distance (85.76% vs. 79.88%). With threshold 1 the precision score of Transliterated Distance was a little bit lower than that of Levenshtein Distance (82.20% vs. 83.94%), but the recall score of Transliterated Distance was higher than that of Levenshtein Distance (95.11% vs. 92.37%).

Revised Roman	Precision(%)	Recall(%)	F-Score
KoTrlitDist[0]	99.86	85.76	0.9227
Levenshtein[0]	99.98	79.88	0.8881

[0] Threshold = 0

Revised Roman	Precision(%)	Recall(%)	F-Score
KoTrlitDist[1]	82.20	95.11	0.8819
Levenshtein[1]	83.94	92.37	0.8795

[1] Threshold = 1

Fig. 3. Performance comparison with Revised Romanization Transliteration

With threshold 0, the F-score for Transliterated Distance is 0.9227 and the F-score for Levenshtein Distance 0.8881, and with threshold 1, the F-score for Transliterated Distance is 0.8819 and the F-score for Levenshtein Distance is 0.8795, consistently showing that the newly developed measure is more effective in finding correct links.

We also tested two cases with each different threshold for McCune-Reischauer Transliteration (an old standard). As summarized in Figure 4, with threshold 0 the precision score of Transliterated Distance (KoTrlitDist in the figure) was almost the same as that of Levenshtein Distance (99.91% vs. 99.97%), but the recall score was much higher than that of Levenshtein Distance (66.61% vs. 46.21%). With threshold 1 the precision score of Transliterated Distance was almost the same as that of Levenshtein Distance (85.00% vs. 86.56%), but the recall score of Transliterated Distance was higher than that of Levenshtein Distance (80.90% vs. 71.93%).

McCune-Reichaur	Precision(%)	Recall(%)	F-Score
KoTrlitDist[0]	99.91	66.61	0.7993
Levenshtein[0]	99.97	46.21	0.6320

[0] Threshold = 0

McCune-Reichaur	Precision(%)	Recall(%)	F-Score
KoTrlitDist[1]	85.00	80.90	0.8290
Levenshtein[1]	85.56	71.93	0.7816

[1] Threshold = 1

Fig. 4. Performance Comparison with McCune-Reischauer Transliteration

Taken together, the results consistently show that the proposed scheme of Transliterated Distance outperforms Levenshtein Distance for recall and is similar to Levenshtein Distance for precision. With threshold 0, the F-score for Transliterated Distance is 0.7993 and the F-score for Levenshtein Distance is 0.6320, and with threshold 1, the F-score for Transliterated Distance is 0.8290 and the F-score for Levenshtein Distance is 0.7816, consistently showing that the newly developed measure is more effective in finding correct links.

4 Summary

The Phoneme Distance measure takes the distribution of phonemes in syllables into account to calculate a distance between two Korean strings. Through multiple empirical testing, we have found out that Korean Phoneme Distance is more useful than Levenshtein Distance in interlinking Korean resources because Korean Phoneme Distance reflects the characteristics of encoding scheme of Korean writing system. This measure is especially effective in enhancing precision by reducing the number of irrelevant records. Through multiple empirical testing, we have also confirmed that the proposed Transliterated Distance measure is much more useful than the Levenshtein Distance measure for interlinking transliterated Korean resources because Transliterated Korean Distance reflects the characteristics of phonetics of Korean. This

measure is especially effective in enhancing recall keeping precision almost intact, thereby contributing to obtaining a higher number of correct links. The two proposed metrics, Phoneme Distance and Transliterated Distance are original, first defined and explained in this paper. The two measurement approaches can be extended to fuse multi-lingual resources as well.

With the new measures, we could control more precisely the range of the target strings while navigating and generating more quality links with regard to Korean resources. While the localization issues described in this paper needs further empirical validation in different language settings, we expect the ideas implemented through those measures to be applicable to many non-Western countries.

Acknowledgements. This research was conducted by the International Collaborative Research and Development Program (Creating Knowledge out of Interlinked Data) and funded by the Korean Ministry of Knowledge Economy.

References

1. Auer, S., Weidl, M., Lehmann, J., Zaveri, A.J., Choi, K.-S.: I18n of Semantic Web Applications. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) ISWC 2010, Part II. LNCS, vol. 6497, pp. 1–16. Springer, Heidelberg (2010)
2. Kim, E., Weidl, M., Choi, K.S., Soren, A.: Towards a Korean DBpedia and an Approach for Complementing the Korean Wikipedia based on DBpedia. In: Proceedings of the 5th Open Knowledge Conference 2010, pp. 1–10 (2010)
3. Volz, J., Bizer, C., Gaedke, M.: Silk – A Link Discovery Framework for the Web of Data. In: WWW 2009 Workshop on Linked Data on the Web, LDOW (2009)
4. Roh, K., Park, K., Cho, H.G., Chang, S.: Similarity and Edit Distance Algorithms for the Korean Alphabet using One-Dimensional Array of Phonemes. The Korean Institute of Information Scientists and Engineers 17, 519–526 (2011)
5. Kang, B., Choi, K.: Automatic Transliteration and Back-Transliteration by Decision Tree Learning. In: LREC 2000 Second International Conference on Language Resources and Evaluation Proceedings, Athens, Greece, pp. 1135–1411 (2000)
6. Jeong, K.S., Myaeng, S.H., Lee, J.S., Choi, K.S.: Automatic Identification and Back-Transliteration of Foreign Words for Information Retrieval. Information Processing & Management 35, 523–540 (1999)
7. Kang, B., Lee, J., Choi, K.S.: Phonetic Similarity Measure for Korean Transliterations of Foreign Words. Journal of Korean Information Science Society 26, 1143–1259 (1999)