Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Language Independent Feature Extractor

Young-Seob Jeong and Ho-Jin Choi
Department of Computer Science, KAIST
291 Daehak-ro, Yuseong-gu, Daejeon 305-701, South Korea
{pinode,hojinc}@kaist.ac.kr

Introduction

We propose a new customizable tool, Language Independent Feature Extractor (LIFE), which models the inherent patterns of any language and extracts relevant features of the language. There are two contributions of this work: (1) no labeled data is necessary to train LIFE (It works when a sufficient number of unlabeled documents are given), and (2) LIFE is designed to be applicable to any language.

Although there are some studies that aim to design language independent feature extractors, we argue that most of them are not truly language independent. First, many works depend on some other resources or tools (e.g., WordNet) which themselves are inherently language-specific. In (Steinberger, Pouliquen, and Ignat 2006), many resources and features were employed, and a huge effort will be required to apply it to other languages. (Curran and Clark 2003) defined word-level features, alphabet-level features, and some features obtained from a gazetteer which again is language-specific. Many of these approaches are available only when these resources are constructed using the target languages.

Second, most of these studies are applicable to alphabet-based languages (e.g., English), but not to non-alphabet-based languages (e.g., Korean, Chinese) because the characteristic difference between the two types of language is not considered. For example, Part-Of-Speech (POS) tags are usually allocated to each morpheme in Korean, while the POS tags in English are allocated to each word(token). There can be no blank space between morphemes or words in Korean and Chinese, while in English, a blank space separates two words. In order to design a truly language independent feature extractor, analysis of documents should not depend on such language-specific assumptions.

There are several approaches (Chen et al. 2010; Jing et al. 2003) that do not depend on such language-specific assumption of word definition. Instead, they employ letter-level features. (Jing et al. 2003) compares letter-level, word-level, and class-level features by their performances in NER task on Chinese language. The class-level features are defined to be the class tags of words such as numbers, Chinese names, foreign names, etc. In this work, the letter-level features provide the best performance. Employing the letter-level features may serve as a good starting point in developing language independent methods. However, trivial letter-level features are often just occurrence-based features, so they cannot convey as much information as POS tags or dependency trees. LIFE generates extended letter-level features which convey such sufficient information.

Class	8	4	5	9	4	7	4	8	3
Topic			0						
Text	W e		a r r i v	e d		a t		n o o n	.

Figure 1: Result of LIFE on a sample sentence written in English.

Fig. 1 shows the result of LIFE applied to a sample sentence written in English. The third row represents the letters of the given sentence which are used as input to LIFE. The first and second rows represent the corresponding class and topic values. These values are the features generated by LIFE. The boxes represent letter sequences, where a letter sequence contains at least one letter. The LIFE outputs a pair of class and topic value for each letter sequence. For an instance, for the letter sequence 'arriv', its class value is 5 and its topic value is 0. The class and topic values are obtained by utilizing a modified version of the probabilistic model used in (Griffiths et al. 2004). In the model, a particular class is assigned to model the topic distributions. In this example, class 5 is assigned to the topic distributions, i.e., the topic value exists only when the class value is 5.

Fig. 2 shows the result of LIFE applied to a sample sentence written in Korean. The meaning of the sentence is '(We) arrived at noon.'. Similar to the Fig. 1, class 5 is assigned to model the topic distributions. The class 5 is allocated to two letter sequences '되어서'(doe-eo-seo) and '도착했'(do-chak-haet), where the two letter sequences have verbal morphemes. Notice that in Fig. 1, the letter sequence 'arriv' is also marked by class 5. We can see that the same class value is allocated to both letter sequences that exhibit similar functionality, i.e. verbal morphemes, despite the fact that the two sentences are written in different languages. The class values represent syntactic features while the topic values represent semantic features. The main difference between a class value and a POS tag is that the class value is allocated to a letter sequence while the POS tag is assigned

Class	8	6	9	4	8	4	5	9	4	5	6	9	3
Topic							0			0			
Text	점	심	때	가		다		되	어	서	야		도	착	했	지	요	.

Figure 2: Result of LIFE on a sample sentence written in Korean.

to morphemes. Although the definition of a word is different in different languages, we can obtain the features from any language because the features are letter-level.

Structure of LIFE

There are three modules in LIFE: Candidate Generator, Document Reader, and Feature Generator. The input of Candidate Generator (CG) module is raw texts without labels. There are two functions in this module: Trie Generator and Combinator. These functions generate candidates of letter sequences. We define a letter sequence as a frequent sequence of letters of tokens, where the length of the letter sequence is at least one letter. There can be different sets of letter sequences with different policies regarding how much of the frequent parts should be split and taken into the letter sequences, and the CG module provides the opportunity to manage such policies by its two functions.

The input of Document Reader (DR) module is the candidate set generated by the Candidate Generator module. The module DR has two functions, Letter Sequence (LS) Dictionary Generator and Corpus Generator. The function Corpus Generator simply goes through all the documents token by token, and divides a token into several letter sequences, and the function LS Dictionary Generator keeps a list of unique letter sequences. These two functions work together to generate an output consisting of a LS dictionary and the documents represented by Bag-Of-'Letter Sequence'(BOLS) scheme.

The input of Feature Generator (FG) module is the LS Dictionary and the documents represented by the BOLS scheme. The module FG has one function, Probabilistic Model, which is a modified version of the probabilistic model used in (Griffiths et al. 2004). Griffiths's model takes documents represented by Bag-Of-Words(BOW), whereas our function takes documents represented by Bag-Of-'Letter Sequences' (BOLS). This is the key point that enables LIFE to be applicable to any language. The module FG outputs a pair of a class value and a topic value for each letter sequence. The class values can be viewed and used as syntactic features, while the topic values as semantic features. The generated pairs can also be utilized in construction of higher level features for further applications.

Time Information Extraction using LIFE

We performed time information extraction using LIFE with the dataset of TempEval-2. We used CRF++ library to extract TIMEX3 extents, and predict their types, with English and Korean datasets. For both languages, we observed that the features obtained from LIFE are better than the features of a base-line model, in which the base-line is the probabilistic model (Griffiths et al. 2004). The performance comparison for Korean is shown in Fig. 3.

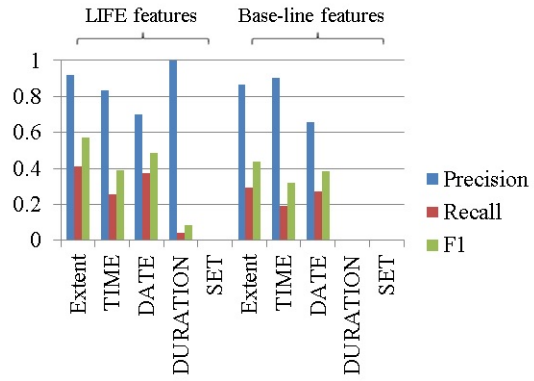

Figure 3: Comparison result for TIMEX3 prediction for Korean, where the vertical axis represents the performance.

We argue that LIFE provides the most language independent features for the following three reasons: (1) it does not require any preprocessing (e.g., stop-word filtering, stemming, morpheme analysis), (2) it does not require labeled dataset, and (3) it generates the features which capture letter-level patterns using the BOLS scheme. We proved the usefulness of LIFE by experimental results of time information extraction. More detailed explanation can be found in https://sites.google.com/site/pinodewaider/home/life.

Acknowledgments

This work was supported by the IT R&D program of MSIP/IITP. [10044577]

References

Chen, Y.; Li, W.; Liu, Y.; Zheng, D.; and Zhao, T. 2010. Exploring deep belief network for chinese relation extraction. In *Proceedings of the CIPS-SIGHAN Joint Conference on Chinese Language Processing*, 113–120.

Curran, J. R., and Clark, S. 2003. Language independent ner using a maximum entropy tagger. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, 164–167.

Griffiths, T. L.; Steyvers, M.; Blei, D. M.; and Tenenbaum, J. B. 2004. Integrating topics and syntax. In *Proceedings of Advances in Neural Information Processing Systems 17*.

Jing, H.; Florian, R.; Luo, X.; Zhang, T.; and Ittycheriah, A. 2003. Howtogetachinesename(entity): Segmentation and combination issues. In *Proceedings of the conference on Empirical methods in natural language processing*, 200–207.

Steinberger, R.; Pouliquen, B.; and Ignat, C. 2006. Exploiting multilingual nomenclatures and language-independent text features as an interlingua for cross-lingual text analysis applications. *Computing Research Repository* abs/cs/0609064(4).