

UTA Stemming and Lemmatization Experiments in the Bengali ad hoc Track at FIRE 2010

A. Loponen* J. Paik† K. Jarvelin

Dept. of Information Studies and Interactive Media
University of Tampere, Finland

ABSTRACT

UTA participated in the monolingual Bengali ad hoc Track at FIRE 2010. As Bengali is highly inflectional, we experimented with three language normalizers: one stemmer, YASS, and two lemmatizers, GRALE and StaLe. YASS is a corpus-based unsupervised statistical stemmer capable of handling several languages through suffixing. GRALE is a novel graph-based lemmatizer for Bengali, but extendable for other agglutinative languages. StaLe is a novel statistical rule-based lemmatizer that has been implemented for several languages. We submitted 6 runs, using YASS for the title-and-description (TD) run, GRALE for the T and TD runs, and StaLe for T, TD and TDN runs, the last one employing also narratives. The T runs were the least effective with MAP about 0.34 (P@10 about 0.30). All the TD runs delivered MAPs close to 0.44 (P@10 about 0.37), while the TDN run gave a MAP of 0.506 (P@10 0.414). The performances of the three normalizers are close to each other, but they have different strengths in other aspects. The performances compare well with the ones other groups have obtained in the monolingual Bengali ad hoc Track at FIRE 2010.

Keywords

Bengali, Stemming, Lemmatization, Information Retrieval

1. INTRODUCTION

UTA participated in the monolingual Bengali ad hoc Track at FIRE 2010. Bengali is a member of Indo-Aryan language family and has several linguistic features that complicate information retrieval. Bengali morphology is very productive. Apart from rich word form inflection, Bengali is productive in compound words, words having more than one root and which can be formed from combinations of nouns, pronouns, adjectives and verbs. Unlike English, Bengali is a relatively free word order language thus complicating syntactic processing, e.g. for ambiguity resolution. Bengali letters also have only one case making it difficult to detect proper nouns. Moreover, words are ambiguous: a proper name can also be an abstract noun or an adjective. Finally, there are little resources for NLP in Bengali.

UTA experimented with three language normalizers: one stemmer, YASS, and two lemmatizers, GRALE and StaLe. YASS is a corpus-based unsupervised statistical stemmer ca-

ble of handling several languages through suffixing. It returns word stems. GRALE is a novel graph-based lemmatizer for Bengali, developed by the second author, and extendable for other agglutinative languages. StaLe is a novel statistical rule-based lemmatizer, developed at UTA, that has been implemented for several languages. Both lemmatizers deliver lemmas, i.e., full dictionary headword forms instead of stems. All normalizers employ a different strategy in word form normalization.

UTA submitted 6 runs to the monolingual Bengali ad hoc Track at FIRE 2010. We experimented with title (T) runs, title-and-description (TD) runs, and title-description-and-narrative (TDN) runs to see the effect of topic length. System-by-system, we used YASS for a TD run, GRALE for T and TD runs, and StaLe for T, TD and TDN runs.

The T runs were the least effective with MAP 0.337 - 0.346 (P@10 from 0.302 to 0.308). All the TD runs delivered MAPs between 0.445 and 0.451 (P@10 from 0.370 to 0.380), while the TDN run gave a MAP of 0.506 (P@10 at 0.414). The performances of the three normalizers are within a narrow band and thus anyone of them could be used. However, they have varying strengths in other aspects. The performances compare well with the ones other groups have obtained in the monolingual Bengali ad hoc Track at FIRE 2010.

The paper is organized as follows: Section 2 briefly presents some features of the Bengali language. Section 3 presents the three normalizers and Section 4 defines the UTA runs. Findings (Section 5), Discussion (Section 6), and Conclusion (Section 7) follow.

2. SOME REMARKS ON THE BENGALI LANGUAGE

Bengali is a highly inflectional language where one root can produce 20 or more morphological variants. Unlike English, proper nouns also can have a number of variations (for example, samir-ke, samir-i, samir-o, samir-er). In most cases variants are generated by adding suffixes to the end of the root. Also two or more atomic suffixes combine to form a single suffix and inflect the root (for example, samir-der-ke-o, where samir is the root, der, ke, o are atomic suffixes). Nouns and pronouns are inflected for case, including nominative, objective, genitive and locative. The case-marking pattern for each noun being inflected depends on the noun's degree of animacy. When a definite article such as -ta (singular) or -gula (plural) is added, nouns also inflect in number. Some of the genitive suffixes are -r, -er, -diger, der. Apart from rich word form inflection, Bengali is productive

*corresponding author's email : aki.loponen@uta.fi

†On leave from Indian Statistical Institute, Kolkata.

in compound words, which can be formed from combinations of nouns, pronouns, adjectives and verbs. There also exist a large number of compound words having more than one root and they have a number of morphological variants. For example the word *dhan* means wealth and *haran* means robbing. These two words combine to form *dhanharan*, meaning robbing of wealth. Now *dhanharan* can produce morphological variants like, *dhanharankari*, *dhanharankarider* where *-kari* and *-der* are suffixes. New words are also formed by derivation. Derivatives and their stems may belong to different parts of speech. Derivatives may change their form significantly from the root word and they are also formed through simple suffixing. For example, the derivation of *madhurjya*, (sweetness) from *modhur* (sweet) is a typical instance of the former kind and *bhadrota* (goodness) from *bhadro* (good) is of the later kind. Derived words and their roots very often use the same suffixes to generate inflectional forms.

Due to these morphological features, word form normalization through stemming or lemmatization is likely to greatly increase the term weights of the normalized forms and therefore to benefit query-document matching.

One typical feature of Indian Languages is that proper names are often either abstract nouns or adjectives. This combined with the fact that Bengali letters have only one case makes it difficult to detect proper nouns. For example *mamata* is a person name and *mamata* means affection. In monolingual information retrieval this may hurt precision for short queries containing a person name as a keyword. The problem is more severe in CLIR from Bengali to other language. Finally, there are little resources for NLP in Bengali.

3. THE UTA EXPERIMENTAL SYSTEMS

3.1 YASS

YASS [1] is a corpus based purely unsupervised statistical stemmer capable of handling a class of languages primarily suffixing in nature. YASS uses a string distance measure to cluster the lexicon such that each cluster expected to contain all the morphological variations of a root word appearing in the corpus. Given two strings X and Y, YASS computes the distance between them. Sufficiently similar strings (by their prefixes) are hierarchically clustered using the complete link clustering approach. YASS delivers stems and these stems are based on the collection seen.

3.2 GRALE

GRALE is a graph-based lemmatizer for Bengali (it can also be adapted to other agglutinative languages). The two-step algorithm in its first step extracts a set of frequent suffixes by measuring their n-gram frequency from the given corpus and then picks up case suffixes manually identified by a native speaker. The words are then considered as a node of a graph and a directed edge from node *u* to *v* exists if *v* can be generated from *u* by addition of a suffix taken from the selected suffix set. The graph built over the lexicon is directed and acyclic. A node might have zero or more in-degree and out-degree. Between any two nodes there may exist more than one path. (An instance where suffix sequences are employed to generate inflectional variants.)

3.3 StaLe

StaLe is a statistical, rule-based lemmatizer developed by Loponen and colleagues [2], that can operate with out-of-vocabulary words as well as with common vocabulary and is easy to adapt into new languages; even languages with scarce linguistic resources. StaLe is based on the TRT transformation rule system by Pirkola et al [3], originally designed for cross-language name and terminology matching.

StaLe has two phases: one-time creation of the transformation rules for a given language, and lemma generation for input words. StaLe creates lemmatization rules from a given corpus of token-lemma pairs where the tokens are inflected. The tokens are extracted from real texts so that the lemmatization rules would represent the language properly. One lemmatization rule is formed from each token-lemma pair by selecting the affixes with a context character from the token and the lemma.

The training data set for Bengali was obtained by randomly selecting 11000 unique inflected tokens from the FIRE 2008 test corpus. The training data set consisted entirely of nouns. Loponen et al. [2] have shown that, in IR applications, performance is not compromised when only nouns are lemmatized. Using only nouns however facilitates rule learning and makes their application more efficient. For each token, a corresponding lemma form was formed by a native Bengali speaker. From the 11 000 token-lemma pairs, a set of 1163 lemmatization rules was generated.

4. THE UTA EXPERIMENTS

4.1 The Test Collection and Search Engine

We used FIRE 2010 Bengali test collection with 50 topics in our experiments. The collection contains 123047 documents with an average of 362 words per document. Average query length for title (T) queries was 6, for TD queries 17, and for TDN queries 44 words. The recall base has 510 relevant documents for 50 topics. As the search engine we used the Lemur toolkit version 4.7 [4], the Indri search engine in particular. The Lemur Toolkit is an open-source toolkit facilitating research in language modelling and information retrieval. The Indri search engine is based on a combination for the language modelling and inference network retrieval frameworks. (See the Lemur toolkit [4])

4.2 The Runs

We conducted six runs for the monolingual Bengali ad hoc Track: TD runs with StaLe, GRALE and YASS; T runs with StaLe and GRALE; and TDN run with StaLe. The runs were conducted by treating the documents and the queries with language normalizers and matching the treated queries and documents. We also formed a baseline run for each topic length (T, TD, TDN) by matching untreated queries and documents.

5. FINDINGS

The following Table 1 shows the MAP and P@10 scores for the runs. In all query lengths T, TD, and TDN the MAP and P@10 improve with language normalization methods over the baseline. On average, normalization improves MAP from the baseline in T queries by 7 percent units and in TD and TDN queries by 6 percent units. In P@10, the normalization methods outperform baseline, on average, by

five percent units in T and TD queries, and by 3 percent units in TDN queries. There is no statistically significant difference between the language normalization methods in the T and TD runs.

Table 1: Run results for StaLe, GRALE, YASS and baseline, and for all query lengths.

		T	TD	TDN
Stale	MAP	33.74	44.88	50.58
	P@10	30.80	37.00	41.40
GRALE	MAP	34.58	44.51	-
	P@10	30.20	37.40	-
YASS	MAP	-	45.11	-
	P@10	-	38.00	-
Baseline	MAP	27.37	38.92	44.55
	P@10	25.20	32.80	38.00

Query query length correlated positively with query performance. The T runs were the least effective with MAP 0.337 - 0.346 (P@10 from 0.302 to 0.308). All the TD runs delivered MAPs between 0.445 and 0.451 (P@10 from 0.370 to 0.380), while the TDN run gave a MAP of 0.506 (P@10 at 0.414).

6. DISCUSSION

The results from our runs show that information retrieval in monolingual Bengali significantly benefits from language normalization. The stemmer has a slight edge over the lemmatizers in the TD runs, while the two lemmatizers perform practically equally well. However, the stemmer's shortcoming is that it can be trained only to stem the training set, while both of the lemmatizers are extrapolative methods and can be applied to words outside from the training corpus.

GRALE was able to handle all word classes while StaLe handled all words as nouns. The results show that despite the exclusion of verbs from the StaLe's training set, StaLe was able to perform on just as high a level as GRALE.

7. CONCLUSION

UTA participated in the monolingual Bengali ad hoc Track at FIRE 2010. We submitted 6 runs, using YASS for the title-and-description (TD) run, GRALE for the T and TD runs, and StaLe for T, TD and TDN runs, the last one employing also narratives. The performances of the three normalizers are close to each other, but they have different strengths in other aspects. YASS handles any collection in a given language, but each application is collection-specific. YASS delivers stems and this is sufficient for monolingual applications but not the best option for CLIR applications. GRALE and StaLe both deliver lemmas, which are convenient for dictionary matching. Moreover, after the learning phase, StaLe is applicable to other, previously unseen collections of a language. None of the normalizers we considered offers compound splitting, which is a desirable property but not a necessity in monolingual applications [5]. The performances compare well with the ones other groups have obtained in the monolingual Bengali ad hoc Track at FIRE 2010.

8. REFERENCES

- [1] Majumder, P., Mitra, M., Parui, S. K., Kole, G., Mitra, P., Datta, K. YASS: Yet another suffix stripper. *ACM Trans. Inf. Syst.*, 25(4):18, 2007
- [2] Loponen, A. Jarvelin, K. A statistical lemmatizer for information retrieval. submitted in 2009.
- [3] Pirkola, A., Toivonen, J., Keskustalo, H., Visala, K., Jarvelin, K. Fuzzy translation of cross-lingual spelling variants. In SIGIR '03': Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval, pp. 345-352, New York, NY, USA, 2003.
- [4] Lemur. The Lemur Tool-kit for Language Modelling and Information Retrieval. <http://www.lemurproject.org/> (visited 11.2.2010).
- [5] Airio, E. Word normalization and decompounding in mono- and bilingual IR. *Information Retrieval* 9: pp. 249-271, 2006.