DCU@FIRE2012: Monolingual and Crosslingual SMS-based FAQ Retrieval

Johannes Leveling
Centre for Next Generation Localisation
School of Computing
Dublin City University
Dublin 9, Ireland
johannes.leveling@computing.dcu.ie

ABSTRACT

This paper presents results for DCU's second participation in the SMS-based FAQ Retrieval task at FIRE. For FIRE 2012, we submitted runs for the monolingual English and Hindi and the crosslingual English to Hindi subtasks. Compared to our experiments for FIRE 2011, our system was simplified by using a single retrieval engine (instead of three) and using a single approach for detecting out-of-domain queries (instead of three). In our approach, the SMS queries are first transformed into a normalized, corrected form. The normalized queries are submitted to a retrieval engine to obtain a ranked list of FAQ results. A classifier trained on features extracted from the training data then determines which queries are out-of-domain and which are not. For our crosslingual English to Hindi experiments, we trained a statistical machine translation system for Hindi to English translation to translate the full Hindi FAQ documents into English. The retrieval then works on the corrected English input and retrieves results from the translated Hindi FAQ documents.

Our best experiments achieved an MRR of 0.949 for the monolingual English subtask, 0.880 for the monolingual Hindi subtask, and 0.450 for the crosslingual subtask.

1. INTRODUCTION

This paper describes the second participation of Dublin City University (DCU) in the SMS-based FAQ Retrieval Task[1] at the Forum for Information Retrieval Evaluation[2] (FIRE). We submitted runs for three subtasks: the English monolingual subtask, the monolingual Hindi subtask, and the crosslingual English to Hindi subtask. The task consists of retrieving the correct answer to an incoming SMS question (possibly in a different language) from a collection of FAQ documents comprising questions and answers on a variety of different topics from career advise to popular Indian recipes. The incoming (English) queries are written in noisy SMS *text speak* or *textese* and contain many misspellings, abbreviations and grammatical errors. SMS queries which have no corresponding answer in the FAQ collection are considered as out-of-domain queries and need to be identified and flagged as out of domain (OOD) by returning "NONE".

The DCU system can be broken down into three distinct phases: SMS normalization, retrieval of ranked results, and identification of out of domain query results [3]. These steps are briefly described in the following section. For our crosslingual experiments, an additional step involving the translation of the full FAQ document collection was added.

Our experiments aim to investigate two aspects of SMS-based FAQ retrieval task 2012:

1. the influence of the out of domain (OOD) detection classifier on system performance;
2. the influence of out of vocabulary (OOV) words in the translated documents for crosslingual experiments.

The rest of this paper is organized as follows: Section 2 introduces the system setup used for the experiments. Section 3 describes the experiments for our submitted runs. Section 4 presents the experimental results before concluding the paper in Section 5.

2. SIMPLIFIED FAQ RETRIEVAL SYSTEM

Our system follows the setup described in [7], which is a simplification of the system described in [3], used for our first participation in this task. The experiments described in this paper are based on the 2012 data for this task. Collection statistics are shown in Table 1.

The first phase in our system involves normalizing words in the SMS text so that they more closely resemble the text in the FAQ data set (i.e. spelling correction and normalization). We employ the same preprocessing steps as described in [3] to correct and normalize spelling in both SMS queries and FAQ documents. As resources for spelling correction, we used a set of frequent spelling errors in Wikipedia articles which were collected from the corresponding Wikipedia page. This set comprises 4,192 spelling error corrections. In addition, we manually generated a set of spelling corrections from the document collection used for the medical record retrieval track at TREC 2012[3]. This set consists of 9,533 corrections of spelling errors, including many run-together words which are split into individual words (e.g. "inthe" would be corrected to "in the").

For the second step in the process, we employed the Lucene retrieval toolkit with our own implementation of the BM25 retrieval model [11] to retrieve a ranked list of candidate answers from the FAQ collection, given the normalized query. The BM25 retrieval model proved to be the single best retrieval approach out of the three approaches used in our previous system [3].

[1] http://www.isical.ac.in/~clia/faq-retrieval/faq-retrieval.html
[2] http://www.isical.ac.in/~clia/
[3] http://trec.nist.gov/data/medical.html

Table 1: Collection statistics for the SMS-based FAQ task.

Language	Documents	Training [all (rel/non_rel)]	Test [all (rel/non_rel)]
English	7251	4476 (3047/1429)	1733 (726/1007)
Hindi	1994	554 (173/381)	579 (200/379)
English to Hindi	1994	554 (173/381)	431 (75/1007)

In the final step, out of domain queries are identified using a classifier based on features extracted during the retrieval process. The detection of out of domain queries is now based solely on a classifier based on TiMBL [1], which was trained on the features described in [3]. The features include the result set size, raw BM25 scores, and the score differences between the top five results. In addition, we used the following features to combine all OOD detection approaches from our previous system into a single classifier:

- The normalized BM25 scores, computed as the maximum possible score for the query q as the sum of IDF scores (see Equation 1) for all query terms.

$$score(q,d) = \sum_{t \in q} \log(\frac{N - df_t + 0.5}{df_t + 0.5}) \quad (1)$$

- The term overlap scores for the SMS query and the top five documents as shown in Equation 2. The term overlap is computed as the number of matches $m(q,d)$ between query q and document d, normalized by the query length $|q|$.

$$overlap(q,d) = \frac{m(q,d)}{|q|} \quad (2)$$

These features were used in the previous version of our system as part of the two additional classifiers. We trained the OOD detection classifier on the full 2012 training data.

3. EXPERIMENT DESCRIPTION

3.1 Monolingual Experiments

For the monolingual English and Hindi experiments, we submitted two runs: one using our OOD classifier (i.e. returning "NONE" when a query was classified as an OOD query) and one not using it (i.e. always returning the retrieved results). Note that even the second approach can return "NONE" when a query consists only of stopwords and no results can be retrieved. We expect that the performance metrics will show that using the OOD detection improves overall system performance. For the monolingual Hindi experiments, we did not employ any spelling correction or normalization method, as we presume that *textese* occurs primarily in the English SMS data.

3.2 Crosslingual Experiments

For the crosslingual English to Hindi experiments, we trained a statistical machine translation (MT) system for translation of the Hindi document collection into English. The system was used to translate the full Hindi FAQ document collection into English, because in a typical information retrieval scenario, Hindi speakers would enter an English query but would prefer to read results in Hindi. Most related work only reports results on translating from English to Hindi (see, for example [2]).

3.2.1 Document Translation

Training a statistical MT system generally includes the steps outlined below.

Data preparation..

We combined parallel training data from different sources to serve as training data. The data preprocessing includes tokenization and normalization of special characters which are based on scripts provided with the Moses toolkit.

Data alignment..

We employ GIZA++[4] to obtain word-aligned parallel training data, using the default settings and the growdiag-final-and symmetrization heuristic [6].

Extracting the phrase table..

The phrase table forming the translation model is obtained from the Moses toolkit[5] [5].

Training a language model..

The language model (LM) for the target language is derived from applying the SRILM toolkit[6] on the combination of all English training data. We create a trigram LM with the Kneser-Ney smoothing method [4].

Tuning..

The translation is tuned based on minimum error rate training (MERT) [9] using separate development data.

Statistical MT for Hindi-English usually suffers from sparse training data. We combined several parallel corpora for our training data. The data we used for the experiments described in this paper consists of the following parallel resources:

- TIDES: The TIDES-IIIT Dataset was originally created for the DARPA-TIDES surprise language contest on Statistical Machine Translation in 2002. It was revised and extended at IIIT Hyderabad and provided for the NLP Tools Contest at ICON 2008[7] [12]. The corpus is general domain with news articles forming the greatest proportion. The training, development and test sets contain 49,504, 988 and 697 sentences respectively.

- EILMT-Tourism Corpus: This dataset is provided by the EILMT consortium funded by DIT, Government of India. It is a domain-specific corpus intended for training machine translation systems for the tourism domain. The training, development and test sets contain

[4] http://giza-pp.googlecode.com/
[5] http://www.statmt.org/moses/
[6] http://www.speech.sri.com/projects/srilm/download.html
[7] http://ltrc.iiit.ac.in/icon2008/nlptools.php

Table 2: Hindi to English translation BLEU scores for different test sets.

Data	Training	Test	Development	BLEU score
TIDES-IIIT	49,504	697	988	13.30
Crowdsourced HI-EN	41,396	8,000	4,000	7.04
ICON	7,000	500	500	25.38

(after de-duplication) 6,755, 500 and 495 sentences, respectively.

- Agro: The English-Hindi-Marathi-UNL parallel corpus from the Resource Center for Indian Language Technology Solutions[8] covers the agricultural domain and contains 527 parallel sentences, of which we used a subset of 246 (correctly aligned) sentences.
- Crowdsourced HI-EN data: A crowdsourced parallel data set available in different Indian languages which is distributed as part of the Joshua decoder[9] [13, 10]. The dataset comprises about 50,000 parallel English-Hindi sentences.
- UWdict: The Universal Word - Hindi Dictionary [10] is a lexical database to aid machine translation and multilingual search. It contains 128,174 translation entries for words, phrases, and acronyms.
- KDE: The English-Hindi KDE data, a subcorpus of parallel data from the the open parallel corpus (OPUS)[11] with 97,227 entries.
- Interlanguage Wikipedia links: Titles of English and Hindi Wikipedia articles, presuming that they correspond to parallel data (27,380 entries).
- FIRE queries: Parallel FIRE queries for the ad-hoc information retrieval task. We extracted 200 topic titles and 200 topic descriptions.

Translation results (measures as BLEU score) on different test corpora are shown in Table 2. The MT system was tuned on the combination of the development sets as listed in the same table. We found that there is a high variance in the BLEU scores for the different test sets. We presume that this might be due to incomplete or incompatible normalization approaches for the different parallel data sets or a potential overlap in some of the data.

3.2.2 *Translating OOV Words*

We aimed to address the problem of data sparsity for Hindi to English translation by combining different parallel corpora for training data to get a high coverage and a low out of vocabulary (OOV) rate. However, we still noticed a high OOV rate in the translated documents after an initial translation run. More than 32K words out of 210K words were left untranslated, i.e. 15.4% are OOV words. We wanted to explore different techniques to reduce the number of OOV words and examine the effect of reducing the OOV rate on the system performance.

To cope with the OOV words, we applied techniques ranging from lookup in lexical resources, translation of compound constituents, to transliteration. We view OOV reduction as a post-processing step in translation. All OOV words are replaced with word translations or transliterations using different techniques. Our OOV reduction is mainly based on two resources for looking up untranslated words: the previously described UWdict, and a manually compiled transliteration lexicon comprising common proper nouns (639 entries).

The main idea for our OOV reduction is to modify the untranslated words and look them (or their constituent parts) up in translation resources. We employed stemming (based on the Lucene Hindi stemmer), normalization, and compound splitting to modify the untranslated words to increase the likelihood of finding a matching dictionary entry with a translation. Normalized terms are obtained by the following character substitution process, similar to the one described in [8]:

- "NA" followed by Virama is replaced with Bindu.
- Chandrabindu is replaced with Bindu.
- Nukta is deleted.
- Zero width joiner characters are deleted.
- Virama is deleted.
- Chandra is replaced with short vowel.
- Long vowels are replaced with their short counterpart.

Compound splitting is based on a greedy approach matching candidate words starting from the left and looking up potential constituent words in the resources. Compound translation succeeds only if all constituent parts could be translated.

Finally, we employed the "Indian languages TRANSliteration" (ITRANS) scheme for Romanization of Hindi words (i.e. words in Devanagari script) to ASCII text resembling English words. The transliteration process is based on tables describing character transliteration[12], followed by application of a few rules for cleaning up the generated output. Table 3 shows the effect of applying the OOV reduction techniques in sequence on the number of OOV words.

4. RETRIEVAL EXPERIMENTS AND RESULTS

All experiments are based on an index of the FAQ document question field, as previous experiments in [3] showed

[8] http://www.cfilt.iitb.ac.in/download/corpus/parallel/agriculture_domain_parallel_corpus.zip
[9] https://github.com/joshua-decoder/indian-parallel-corpora/tree/master/hi-en
[10] http://www.cfilt.iitb.ac.in/~hdict/webinterface_user/index.php
[11] http://opus.lingfil.uu.se/
[12] http://en.wikipedia.org/wiki/Devanagari_transliteration

Table 3: Effect of eliminating OOV words in translated output of 210,925 words. 31,788 words (15.4%) were initially left untranslated.

Method	Lookup form	Lookup data	Count	% Reduction
1	original term	dictionary	4,728	(14.5%)
2	original term	transliterations	83	(0.3%)
3	normalized term	dictionary	419	(1.3%)
4	normalized term	transliterations	24	(0.1%)
5	stemmed term	dictionary	1,413	(4.4%)
6	stemmed term	transliterations	13	(0.0%)
7	stemmed normalized term	dictionary	135	(0.4%)
8	stemmed normalized term	transliterations	0	(0.0%)
9	compound constituents	dictionary	721	(2.2%)
10	transliteration	N/A	24,973	(76.8%)
		Σ	32,509	

that this yielded the best results compared to indexing the answer fields or a combination of the FAQ question and answer fields. For the crosslingual experiments, we index the translated FAQ document question field.

Results for our submitted runs are shown in Table 4. For monolingual English experiments, adding OOD detection decreases results, but increases MRR. Without OOD detection, only a few out of domain queries are found. For monolingual Hindi experiments, less than half of the in domain queries could be answered. Adding OOD detection results in finding almost all OOD queries, but reduces the number of correct ID queries considerably. For our crosslingual English to Hindi experiments, a similar result can be observed. In addition, reducing the number of OOV words in the translated documents actually decreases performance slightly. Interestingly, many untranslated words in the original form can be found in UWdict, one of the resources used to train the statistical MT system.

5. CONCLUSION AND FUTURE WORK

Our experiments for the SMS-based FAQ Retrieval task 2012 use a much simpler version of the system we developed for our participation in 2011. We still achieved a good performance for the monolingual English task. For this task, the OOD detection actually improved system performance, possibly because there is enough training data for the OOD classification. For the monolingual Hindi experiments, adding OOD detection reduces the number of correct in domain answers, but covers most OOD queries. The performance for monolingual Hindi and the crosslingual experiments is lower compared to the monolingual English, which might be due to a missing correction and normalization process. Some results could be explained by unbalanced training data, i.e. there is a bias in the training instances towards OOD queries.

We plan to analyze in detail the effect of the different stages in the OOV reduction phase and if there is a need to use spelling correction and normalization for Hindi.

6. ACKNOWLEDGMENTS

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (CNGL) project. Special thanks to Debasis Ganguly for his support in the Hindi transliteration.

7. REFERENCES

[1] Walter Daelemans, Jakub Zavrel, Ko van der Sloot, and Antal van den Bosch. TiMBL: Tilburg memory based learner, version 6.2, reference guide. Technical Report 09-01, ILK, 2004.

[2] Rejwanul Haque, Sudip Kumar Naskar, Josef van Genabith, and Andy Way. Experiments on domain adaptation for English-Hindi SMT. In Olivia Kwong, editor, *PACLIC*, pages 670–677. City University of Hong Kong Press, 2009.

[3] Deirdre Hogan, Johannes Leveling, Hongyi Wang, Paul Ferguson, and Cathal Gurrin. DCU@FIRE 2011: SMS-based FAQ retrieval. In *FIRE 2011, 3rd Workshop of the Forum for Information Retrieval Evaluation, 2-4 December, IIT Bombay*, pages 34–42, 2011.

[4] Reinhard Kneser and Hermann Ney. Improved backing-off for m-gram language modeling. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASP)*, volume 1, pages 181–184. IEEE Computer Society Press, 1995.

[5] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In John A. Carroll, Antal van den Bosch, and Annie Zaenen, editors, *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*. ACL, 2007.

[6] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *NAACL 03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT-NAACL)*, pages 48–54. ACL, 2003.

[7] Johannes Leveling. On the effect of stopword removal for sms-based faq retrieval. In Gosse Bouma, Ashwin Ittoo, Elisabeth Métais, and Hans Wortmann, editors, *Natural Language Processing and Information Systems - 17th International Conference on Applications of Natural Language to Information Systems, NLDB 2012, Groningen, The Netherlands, June 26-28, 2012.*

Table 4: Results on the 2012 test data.

		Parameters				
Run	Language	OOD detection	OOV reduction	ID Correct	OOD Correct	MRR
1	English	N	-	661/726 (0.910)	19/1007 (0.019)	0.937
2	English	Y	-	595/726 (0.820)	981/1007 (0.974)	0.949
1	Hindi	N	-	77/200 (0.385)	13/379 (0.034)	0.473
2	Hindi	Y	-	26/200 (0.130)	375/379 (0.693)	0.880
1	Cross	N	N	29/75 (0.387)	41/1007 (0.041)	0.450
2	Cross	N	Y	22/75 (0.293)	60/1007 (0.060)	0.365
3	Cross	Y	Y	4/75 (0.053)	989/1007 (0.982)	0.444

Proceedings, volume 7337 of *LNCS*, pages 128–139. Springer, 2012.

[8] Johannes Leveling and Gareth J. F. Jones. Sub-word indexing and blind relevance feedback for English, Bengali, Hindi, and Marathi IR. *ACM Transactions on Asian Language Information Processing (TALIP)*, 9(3), September 2010.

[9] Franz Josef Och. Minimum error rate training in statistical machine translation. In Erhard W. Hinrichs and Dan Roth, editors, *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, 7-12 July 2003, Sapporo Convention Center, Sapporo, Japan*, pages 160–167. ACL, 2003.

[10] Matt Post, Chris Callison-Burch, and Miles Osborne. Constructing parallel corpora for six indian languages via crowdsourcing. In *Seventh Workshop on Statistical Machine Translation*, pages 401–409, Montréal, Canada, June 2012. ACL.

[11] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline M. Hancock Beaulieu, and Mike Gatford. Okapi at TREC-3. In Donna K. Harman, editor, *Overview of the Third Text Retrieval Conference (TREC-3)*, pages 109–126, Gaithersburg, MD, USA, 1995. National Institute of Standards and Technology (NIST).

[12] Sriram Venkatapathy. NLP tools contest – 2008: Summary. In *ICON 2008 NLP Tools Contest*. Pune India, 2008.

[13] Omar F. Zaidan and Chris Callison-Burch. Crowdsourcing translation: Professional quality from non-professionals. In *49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1220–1229, Portland, Oregon, USA, June 2011. ACL.