Mach Translat (2018) 32:105–126
https://doi.org/10.1007/s10590-017-9205-3

The ARIEL-CMU situation frame detection pipeline for LoReHLT16: a model translation approach

Patrick Littell[1] · **Tian Tian**[1] · **Ruochen Xu**[1] · **Zaid Sheikh**[1] · **David Mortensen**[1] · **Lori Levin**[1] · **Francis Tyers**[2] · **Hiroaki Hayashi**[1] · **Graham Horwood**[3] · **Steve Sloto**[1] · **Emily Tagtow**[1] · **Alan Black**[1] · **Yiming Yang**[1] · **Teruko Mitamura**[1] · **Eduard Hovy**[1]

Received: 31 May 2017 / Accepted: 3 October 2017 / Published online: 27 October 2017
© Springer Science+Business Media B.V. 2017

Abstract The LoReHLT16 evaluation challenged participants to extract Situation Frames (SFs)—structured descriptions of humanitarian need situations—from monolingual Uyghur text. The ARIEL-CMU SF detector combines two classification paradigms, a manually curated keyword-spotting system and a machine learning classifier. These were applied by translating the *models* on a per-feature basis, rather

✉ Patrick Littell
plittell@cs.cmu.edu

Tian Tian
ttian@andrew.cmu.edu

Ruochen Xu
ruochenx@andrew.cmu.edu

Zaid Sheikh
zsheikh@cs.cmu.edu

David Mortensen
dmortens@cs.cmu.edu

Lori Levin
lsl@cs.cmu.edu

Francis Tyers
ftyers@prompsit.com

Hiroaki Hayashi
hiroakih@cs.cmu.edu

Graham Horwood
graham.v.horwood@leidos.com

Steve Sloto
ssloto@gmail.com

Emily Tagtow
etagtow@cs.cmu.edu

than translating the input text. The resulting combined model provides the accuracy of human insight with the generality of machine learning, and is relatively tractable to human analysis and error correction. Other factors contributing to success were automatic dictionary creation, the use of phonetic transcription, detailed, hand-written morphological analysis, and naturalistic glossing for error analysis by humans. The ARIEL-CMU SF pipeline produced the top-scoring LoReHLT16 situation frame detection systems for the metrics SFType, SFType+Place+Need, SFType+Place+Relief, and SFType+Place+Urgency, at each of the three checkpoints.

Keywords LoReHLT · LORELEI · Situation frames · Information extraction

1 Introduction

1.1 Task description

Situation frames (SFs) are semantic structures intended to "enable information from many different data streams to be aggregated into a comprehensive, actionable understanding of the basic facts needed to mount a response to an emerging situation" (Strassel et al. 2017). Each situation frame represents information relevant to humanitarian mission planning, including the type of humanitarian need (e.g., food, water, evacuation) or local background issue (e.g., terrorism or widespread crime), the location of the need or issue, whether the need is current and/or urgent, and whether the need has been partially or fully addressed. Being able to extract this information rapidly and reliably from text (such as local newspaper articles, social media posts, and text messages) may support humanitarian disaster relief planners.

Unfortunately, we are not in a position to apply current natural language processing (NLP) technology directly to this problem. Roughly half of humanity speaks a language that is not one of the top 20 languages by number of speakers (Lewis et al. 2015), and even some of those top 20 languages still count as low-resource languages for the purposes of NLP. Most languages on earth—even languages with tens of millions of

Alan Black
awb@cs.cmu.edu

Yiming Yang
yiming@cs.cmu.edu

Teruko Mitamura
teruko@cs.cmu.edu

Eduard Hovy
hovy@cmu.edu

[1] Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, USA

[2] School of Linguistics, National Research University «Higher School of Economics», Moscow, Russia

[3] Leidos, Inc., Reston, VA, USA

Table 1 An example sentence from document IL3_WL_031228_20140215_G0040006G, and the reference situation frame that systems are attempting to predict

Original sentence	ئاتچان يېزىلىق ھۆكۈمەت بۇ قېتىملىق يەر تەۋرەشتە بەختكە يارىشا بىرمۇ ئادەمنىڭ يارىلانمىغانلىقى، ئەمما ئۆيلەرگە دەز كەتكەنلىكىنى ئېيتقان (The Aqqan Township government said, in later reports, that there were fortunately no casualties, but the quake had caused some housing damage)
SFType	Shelter
Place	Aqqan Township
Need	Current
Relief	No known resolution
Urgency	Non-urgent

speakers—do not even have rudimentary NLP pipelines in place, and the typical NLP pipeline for a language has, in the past, taken years of development.

These challenges—a challenging classification and information extraction task, on challenging languages, within an emergency timespan—come together in the LoReHLT[1] evaluation, in which participants received a surprise language (in 2016, Uyhgur, a Turkic language of the Xinjiang province of China) and a surprise incident (the February 2014 earthquake in Hotan, Xinjiang, China).

Participants had one month to develop a system that detects situation frames in Uyghur monolingual text, evaluated at three checkpoints: 7 days, 14 days, and 28 days. An example sentence and its associated situation frame are illustrated in Table 1.

In a novel addition to NIST evaluations, participants had access to Native Informants (NIs) for a limited number of hours per checkpoint (one hour per task before the first checkpoint, and four additional hours per task after the first checkpoint). These NIs could be utilized in many ways—translating data, doing validation and/or error analysis on system outputs, brainstorming keywords and features—but teams were not permitted to show the NI (or any other human analyst) the evaluation data itself. This was to ensure that, while human expertise could be used to engineer and train the systems in the first place, the systems could still detect situation frames independent of the input of native speakers, linguists, and other subject-matter experts.

1.2 The ARIEL-CMU approach

Given the lack of annotated data in the incident language, it was necessary to transfer from a better resourced language for which some humanitarian need/issue classifications exist (in this case, English, as detailed in Sect. 2.3). This transfer might occur via three basic routes:

[1] www.nist.gov/multimodal-information-group/lorehlt-2016-evaluations.

1. By developing an English classifier model, translating the Uyghur text into English, and applying the classifier to the result.
2. By developing an English classifier model, translating the model’s features (that is, words and short phrases) into Uyghur, and applying the resulting model to untranslated Uyghur.
3. By developing an English classifier model, classifying the English side of a Uyghur-English parallel text, transferring the labels to the Uyghur sentences, training a new Uyghur classifier model on the labeled Uyghur sentences, and applying it to untranslated Uyghur.

The ARIEL-CMU SF team took the second route (and, to our knowledge, was the only team to take this approach).

The model translation approach made the ARIEL-CMU system dependent primarily on *lexical resources*, which are easier to supplement in data-poor situations or situations where the data is out of domain (Sect. 2.4), whereas the text translation and label transfer approaches leave systems dependent on *parallel text*, which is much more difficult to supplement. The ARIEL-CMU system was thus relatively unaffected by the very out-of-domain parallel text included in the training set (consisting mostly of Qur’an translations and software localization files). Lexical resources have other advantages over parallel text, too: for example, they are relatively amenable to human manipulation. This attribute proved to be another key advantage for the ARIEL-CMU system. Finally, this approach allowed the SF team to start work immediately, without having to wait for a Uyghur-to-English translation module to be built by the ARIEL-CMU MT team.

The ARIEL-CMU SF detection pipeline is shown in Fig. 1. Each classifier model (Sects. 3.1, 3.2) consumes lemmatized (Sect. 2.5.2) Uyghur text, English-Uyghur lexicons (Sects. 2.1, 2.4), and classified English humanitarian assistance and disaster relief (HA/DR) text (Sect. 2.3), and produces a list of hypothesized SFs for each sentence. The location field of each SF is then linked using the output of the ARIEL-CMU named entity recognition pipeline (Sect. 4), and then SFs that were not found by both classifiers are removed (Sect. 3.3). Finally, we fill the Status fields (Sect. 5) and run special-case post-filters (Sect. 6.3).

1.3 Evaluation

Systems submitted to the LoReHLT16 competition were scored officially by a novel metric called ‘Situation Frame Error’ (SFE), the sum of false positives and false negatives, divided by the number of reference situation frames. This metric was calculated five times at each checkpoint, for a variety of combinations of result types:

- *SFType* Whether the Type of the humanitarian need (one or more of: Water, Food, Shelter, Medical, Evacuation, Infrastructure, Search/Rescue, Utilities/Energy/Sanitation) or background issue (one or more of: Terrorism or Extreme Violence, Civil Unrest or Widespread Crime, Regime Change) is correct.
- *SFType+Place* Whether both the Type and Place of the need/issue are correct; the possibilities for Place are any relevant named entity (LOC or GPE) in the same document.

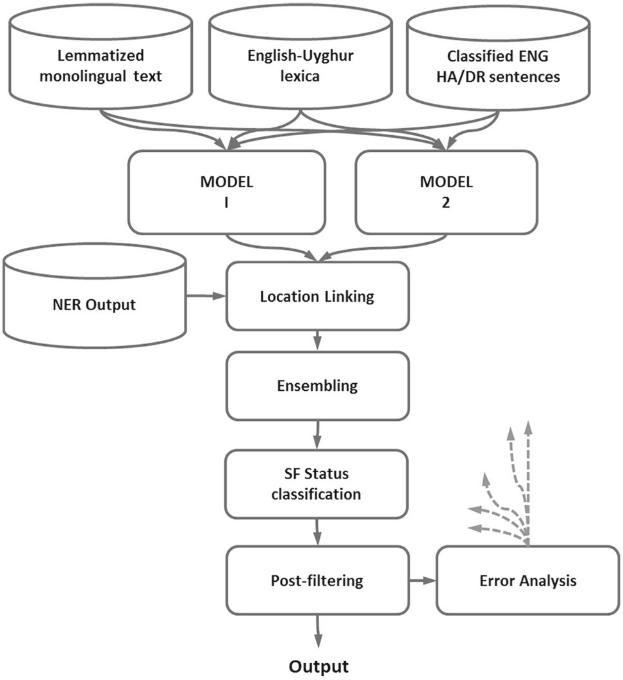


Fig. 1 The ARIEL-CMU situation frame extraction pipeline

- *SFType+Place+Need* Whether the Type, Place, and time status (Current, Past Only, Future Only) of the need are correct.
- *SFType+Place+Relief* Whether the Type, Place, and relief status (Sufficient relief, Insufficient relief or Unknown sufficiency, No relief known) of the need are correct.
- *SFType+Place+Urgency* Whether the Type, Place, and urgency status (Urgent, Non-urgent) of the need are correct.

1.4 Results

Of the five primary systems submitted in the exercise, the ARIEL-CMU system had the best (that is, lowest) SFE rate at each checkpoint, for each of the metrics except for SFType+Place, in which the ARIEL-CMU system had the second-best SFE rate.

SFE is heavily weighted towards precision,[2] and many of our decisions in the pipeline (for example, the decision to intersect the results of the two SF extraction engines as an ensembling method in Sect. 3.3) aimed at the goal of higher precision. The ARIEL-CMU systems had the highest precision scores for SFType.

However, the system's precision did not detract from its F1; the ARIEL-CMU system also had the best F1 measure for SFType. We were again second-best for

[2] It should be noted, however, that it does not vary monotonically with precision (or recall or F1), as the SFE and precision values in Tables 5 and 6 will show.

Table 2 SFE, P, R, and F1 scores on SFType and SFType+Place, for our primary systems at each checkpoint

Chkpt.	SFType				SFType+Place			
	SFE	P	R	F1	SFE	P	R	F1
CP1	1.082	0.424	0.226	0.295	**1.405**	0.104	0.053	0.070
CP2	**1.042**	**0.464**	**0.273**	**0.343**	1.666	0.072	0.056	0.063
CP3	1.147	0.385	0.245	0.300	1.675	**0.121**	**0.107**	**0.113**

Bold values indicate our best performance in a primary submission (highest score for P, R, and F1, lowest score for SFE)

SFType+Place; comparative P/R/F1 measures for the other three metrics, or for contrastive systems, were not reported.

We, like some other performers, found our third-checkpoint systems underperforming on SFType classification compared to our second-checkpoint systems. We believe this to be largely because of overgeneration (especially when considering the SFE metric). At earlier checkpoints, as the classifiers consider text in which they can recognize few words, they also generate few situation frames. By checkpoint 3, our data preprocessing pipeline (Sect. 2.5) had improved significantly, and meanwhile the classifiers were now considering multiword phrases rather than just single words (Sects. 3.1, 3.2), meaning that systems were able to identify more meaningful elements in each sentence, and consequently generated additional situation frames, and with that additional spurious situation frames.

Meanwhile, however, our SFType+Place scores continued to rise (this is easier to see in the P/R/F1 values in Table 2 than in SFE, which has a sometimes chaotic relationship to other metrics), largely because of a substantial improvement in the upstream named-entity recognition pipeline (Bharadwaj et al. 2016).

Our scores for the Status fields (Need, Relief, and Urgency) are considered in Sect. 5 and Table 6.

2 Data and resources

2.1 Uyghur data

We made use of many resources from LDC2016E57 (LORELEI IL3 Incident Language Pack for Year 1 Eval) (Strassel and Tracey 2014). The most crucial resource, for the purposes of our SF pipeline, was the Uyghur-English dictionary; the next most crucial were the Uyghur-Mandarin dictionary and the Uyghur-English parallel text (which we used to generate additional bilingual lexicons in Sect. 2.4).

We made use of the LDC2016E57 monolingual Uyghur text to train Uyghur word vectors (Sect. 3.2).

We also made use of all three Uyghur reference grammars provided in LDC2016E57, in building orthographic converters (Sect. 2.5.1), in building rule-based Uyghur morphological parsers (Sect. 2.5.2), and in training linguistic analysts (Sect. 6).

2.2 Mandarin data

In addition to using the Uyghur-Mandarin dictionary in LDC2016E57, we also made use of the CEDICT Mandarin-English dictionary[3] (included in LDC2016E30, LORELEI Mandarin Incident Language Pack V2.0); this was used to derive a Uyghur-English lexicon in Sect. 2.4.

2.3 English data

2.3.1 English monolingual HA/DR corpora

A variety of in-domain, monolingual corpora were used in the evaluation. Raw documents were collected from the following sources.

ReliefWeb A corpus of more than 491 K documents, totaling over 300 M words, was collected from ReliefWeb,[4] an aggregator of HA/DR news and analysis sponsored by the United Nations Office for the Coordination of Humanitarian Affairs. Approximately 28% of articles are annotated (by ReliefWeb) for one or more disaster types and a disaster name (e.g., ‘Myanmar: Tropical Cyclone Nargis—May 2008’). Just under half (45%) of the articles are also annotated for a humanitarian assistance ‘theme’ (e.g., ‘Health’, ‘Water Sanitation Hygiene’, etc.).

Crisis.net An online collection of global crisis data from various social media sources.[5] Collected content was tagged for ‘disaster’ and ‘conflict’, completed 4/12/2016, and was further limited to data from Ushahidi[6] (~30 K words), Facebook (~58 K words), and Twitter (~415 K words).

Open Source Enterprise (OSE) A set of 24 K articles collected from the Open Source Enterprise[7] portal with a set of disaster keyword queries. All of the documents are natively annotated for one or more topic areas, many of which are of relevance to LORELEI (‘aid’, ‘terrorism’, etc.).

These data sources were used to derive a collection of more topically-focused data sets.

2.3.2 HA/DR topic lexicon

A semi-supervised method was used to construct a set of ~ 34 K terms (words and multi-word expressions) organized around 25 ReliefWeb disaster categories and general situation types proposed by the LORELEI community in early stages of the

[3] http://cc-cedict.org.

[4] http://reliefweb.int.

[5] http://crisis.net.

[6] www.ushahidi.com.

[7] www.opensource.gov.

program: Cold Wave, Cyclone, Drought, Earthquake, Energy, Evacuation, Flood, Food, Heatwave, Infestation, Intervention, Landslide, Medical, Money, Politics, Rescue, Sanitation, Services, Shelter, Terrorism, Tsunami, Civil Unrest, Utilities, Water, and Wildfire. A set of seed terms for each defined topic area was constructed manually with the input of a domain expert; additional terms from CrisisLex's CrisisLexRec (Olteanu et al. 2014) and EMTerms (Temnikova et al. 2015) lexicons were included in these sets where appropriate.

For each seed set, candidate terms were generated with an ensemble of word embedding models (Mikolov et al. 2013a), including: pre-trained Google News word2vec vectors; a word2vec model trained on over one billion English language tweets; a word2vec model trained on the ReliefWeb and OSE data; a singular-value decomposition of dependency path features constructed from the ReliefWeb and OSE data; and a latent semantic indexing model of an English language thesaurus. For each topic, two thousand most topically-similar terms were selected as candidates for manual auditing; terms were ranked by average cosine similarity relative to seed set centroid vectors. After first filtering terms to remove commonly occurring personal and place names, non-ASCII terms, terms of three or fewer characters, and terms with non-word punctuation, the candidate terms were audited with CrowdFlower.[8]

Contributors were asked to rate each term's relevance to the topic on a five point scale, with extreme points on the scale described as indicating a-contextual relevance (e.g., 'sewage' is necessarily relevant to Sanitation without any additional context) or irrelevance (e.g., it is difficult to imagine how 'bubblegum' would be relevant to Extreme Violence/Terrorism), and the mid-range indicating contextual dependence (e.g., 'water' can be relevant to a discussion of Energy in the context of hydroelectricity plants). Terms receiving an average relevance of 3.5 or lower were dropped from the final lexicon. Overall agreement among participants on the rating scale was 75%.

2.3.3 HA/DR topic example sentences

From the ReliefWeb and Crisis.net corpora, we created a collection of 163K topically-categorized example sentences, The sentences were extracted by simple retrieval of selected terms in the HA/DR Topic Lexicon: terms with a high (greater than 4 in a 1–5 scale) human-judged topic relevance and a word length greater than one.

2.3.4 Additional derived data

We also collected an English corpus ahead of the evaluation period by searching for a small set of HA/DR keywords (e.g., 'evacuation', 'earthquake'), and taking the first several thousand articles returned as a categorized English HA/DR corpus.

During the evaluation period, we also annotated a small number of English articles with situation frame types, using the BRAT annotation tool (Stenetorp et al. 2012). This data was used to tune the English SFType classifier (Sect. 3.2).

[8] www.crowdflower.com.

2.4 Deriving bilingual lexicons

As described below, both extraction models (Sect. 3.1, Sect. 3.2) rely primarily on bilingual lexicons to achieve model translation, so the completeness and correctness of the lexicon with respect to the target domain is a matter of critical importance.

Fortunately, it is easier to create parallel *lexicons* than it is to create parallel *text*; we created two additional lexicons to supplement the Uyghur-English lexicon included in LDC2016E57 as follows.

First, we ran `fast_align` (Dyer et al. 2013), a reparameterization of IBM Model II (Brown et al. 1993), on the LDC2016E57 parallel text to derive word-level alignments, then selected all word pairs above a certain frequency threshold. Those words that we knew to be of interest to our extractors were then presented to the native informant for validation (Sect. 6.1).

Second, we took the transitive closure of the Uyghur-Mandarin lexicon included in LDC2016E57 and the Mandarin-English lexicon CEDICT (also included in LDC-2016E30) to build a Uyghur-English lexicon.

Both of these derived lexicons were much richer in single-word translations than the Uyghur-English dictionary provided in LDC2016E57. The latter dictionary contained more phrase-to-phrase translations, which turn out to be less directly useful for model translation.

Finally, we asked the native informant to translate into Uyghur some additional HA/DR-related terms that we did not find in other resources.

2.5 Data preprocessing

2.5.1 Orthographic conversion

Many ARIEL-CMU modules (including the SF, machine translation, and named-entity recognition systems) use the International Phonetic Alphabet (IPA) as their internal representations of text. Having a standardized representation allows us to develop standard tooling (e.g., phonetic search tools) and phonetics-based cross-linguistic models without the need to adapt each tool to each new language.

Orthography-to-IPA conversion (a type of G2P, or grapheme-to-phoneme conversion) in the ARIEL-CMU pipeline is handled by the Epitran module,[9] which uses conversion tables, preprocessing rules, and postprocessing rules to convert native orthography into IPA wherever possible. This requires some manual rule engineering, but it usually requires fewer than four person-hours of work to adapt the Epitran system to a new language. The large number of languages and scripts already supported (currently 62 language-script pairs, with 13 different scripts) means that this manual work is often a matter of adapting support from a related language or script.

For Uyghur orthography-to-IPA conversion, we made use of the Uyghur grammar books in the LDC2016E57 language pack, which contained tables and usage examples for Uyghur orthography. Uyghur orthography-to-IPA conversion was relatively

[9] http://github.com/dmort27/epitran.

straightforward, when compared to conversion of other Perso-Arabic scripts, because the Uyghur orthography fully and unambiguously indicates vowels, whereas other Perso-Arabic scripts tend to have some unwritten or ambiguous vowels.

2.5.2 Lemmatization

We tried three different approaches to lemmatization, each based on hand-written rules but differing in their parsing paradigm.

Regular expression-based morphlogy First, we made a quick lemmatizer (named in our submission filenames as ‘dummorph’, for ‘dummy morphology’), derived from the Uyghur grammar books in the LDC2016E57 language pack, operating directly on Uyghur’s Perso-Arabic orthography, in which hand-written suffixation rules are compiled to a regular expression which captures only the lemma. This system had the benefit of being straightforward to code and specify, and trivial to integrate into our systems, but the downside that it could only output a single hypothesis as the lemma, which (as shown by manual inspection of the output) was often under- or over-lemmatized. All of our Checkpoint 1 submissions used this lemmatizer.

We also used regular-expression-based morphology to perform some limited lemmatization of the LDC2016E57 Uyghur-English lexicon, in particular removing the citation form suffix *-maq* and its variants from verbs, since these are relatively uncommon in text.

Finite-state morphology Our second lemmatizer (named in our submission filenames as ‘franmorph’, for ‘Francis’s morphology’) was adapted from the Uyghur finite-state morphological parser[10] in the Apertium MT toolkit (Forcada et al. 2011), further developed by its author to include additional roots and a root ‘guesser’ during the LoReHLT16 evaluation. The system was developed using the Helsinki Finite State Toolkit (Linden et al. 2011) according to the methods described in Washington et al. (2014).

The benefit of the resulting lemmatizer was that it was based on a much more sophisticated understanding of Uyghur morphophonology (e.g., the raising of front vowels to [i]) compared to our other lemmatizers; the downside was that it was more difficult to integrate into our pipelines than the other lemmatizers (which, like the rest of our systems, were written in Python). Table 3 gives some example output from this system.

Parser-combinator morphology Our third lemmatizer (named in our submission filenames as ‘ipamorph’, since it operates on the output of our Epitran IPA system) arose from a refactoring of the regular-expression compiler to be a recursive descent parser, by converting the primitive elements (e.g., suffix specifications) to parser combinators (Hutton and Meijer 1988; Frost and Launchbury 1989) rather than regular expression snippets. We also significantly expanded the suffix inventory and morphotactic com-

[10] http://svn.code.sf.net/p/apertium/svn/incubator/apertium-uig.

Table 3 Example output from *franmorph* including lemmas and detailed lexical and morphological information

Surface form	Lemma	Tags	Gloss
سىياسەت	سىياسەت	`n.nom`	*policy*
ۋە	ۋە	`cnjcoo`	*and*
تەدبىرلەر	تەدبىر	`n.pl.nom`	*measures*
قانچە	قانچە	`det.itg`	*how*
ياخشى	ياخشى	`adj`	*good*
بولغىنى	بول	`v.iv.ger_past.px3sp.nom`	*being*
بىلەن	بىلەن	`post`	*with*
ياخشى	ياخشى	`adj`	*good*
ئىجرا	ئىجرا	`n.nom`	*implementation*
قىلىنمىسا	قىل	`v.tv.pass.neg.gna_cond.p3.sg`	*if is not done*
ئەھمىيىتى	ئەھمىيەت	`n.px3sp.nom`	*importance*
بولمايدۇ	بول	`v.iv.neg.aor.p3.sg`	*is not*
.	.	`sent`	.

In this example sentence, spurious analyses have been excluded in the interest of readability. The sentence means "However good policy and preventative measures are, if a good implementation is not done they are not of importance"

Table 4 Sample outputs from the IPAMorph multiple-output morphology system (Sect. 2.5.2)

Original word	كىشىلەرنىڭ
IPA	kiʃilɛrniŋ
Lemma	kiʃi
Breakdown	kiʃi-lɛr-niŋ
Gloss	kiʃi-PL-GEN
Naturalistic gloss	of several people

plexity as the checkpoints progressed, based again on the Uyghur grammar books in the LDC2016E57 language pack.

The flexibility of parser combinators with respect to outputs (Hutton 1992) and element reordering (since they produce recursive descent parsers, rather than finite-state automata) allowed us to make a multiple-output parser that simultaneously produced lemmas (for the SF pipeline), morphological segmentations and glosses (for the machine translation pipeline), and a human-readable naturalistic gloss (for human data and error analysis). A sample of IPAMorph outputs are shown in Table 4.

The downside of IPAMorph is that it does not include the range of phenomena in the Apertium FST, nor the more sophisticated morphophonology, the elegant integration of morphology with phonology being a strength of FST-based systems (Beesley and Karttunen 2003).

2.5.3 Morphological disambiguation

In the absence of gold-standard morphological breakdowns, we chose between competing morphological parses using a variety of heuristic penalties.

Most important, a significant penalty is assessed if the morphological parse does not result in a lemma that we can find in one of our Uyghur-English bilingual lexicons (After all, our models were translated, on a word-feature-by-word-feature basis, through these Uyghur-English lexicons, so the fundamental purpose of lemmatization within our SF pipeline is to render the data in the exact form that these models can recognize).

When multiple parses resulted in known lexical items, we decided between them based on the probability (according to an English bigram model) of their English definitions. For example, قاتارلىق (*qatarliq*) could have the lemma *qatarliq* (meaning ‘including’, ‘like’, ‘such as’, ‘et cetera’) or the lemma *qatar* (meaning ‘row’, ‘line’, ‘Qatar’, or ‘board game called 方 in Chinese’). Each hypothetical parse receives an additional penalty according to the negative log probability of its English translation (so the parse resulting in *qatar* receives a higher penalty than *qatarliq*).[11]

Additional, smaller penalties accrue for the length of the lemma (so that shorter lemmas are preferred) and number of suffixes (so that the system avoids choosing a hypothesis with more suffixes when fewer are possible).

This disambiguation system was used for both the Apertium and IPAMorph parsers.

The integration of more sophisticated lemmatization and heuristic disambiguation was the primary difference between our Checkpoint 1 (using ‘dummorph’) and Checkpoint 2 (using IPAMorph) primary systems; as seen in Table 2 this contributed to a small improvement in SFE and a more substantial improvement in F1 score.

Using the Apertium parser gave very similar downstream results; the best system using IPAMorph (our CP2 primary system) outperformed the best system using Apertium (one of our CP2 contrastive systems) by only a single situation frame (out of hundreds).

2.5.4 Naturalistic glossing

One of the outputs of the IPAMorph system, ‘naturalistic glossing’, played a significant role in ARIEL-CMU data and error analysis. Taking advantage of the exclusively suffixing nature of Uyghur and the Mirror Principle (Baker 1985) (that linear order of suffixes usually reflects their syntactic derivation), we mirrored rough English equivalents of each morpheme to the other side of the lemma (e.g., transforming ‘people-PL-GEN’ to ‘of several people’). While the resulting outputs are not nearly as fluent as the output of a more sophisticated machine translation system, it affords a non-speaker analyst a view of the data that is usefully close to the original (as it cannot drop or hallucinate words to match an English language model, and can only perform very limited word rearrangements).

As our analysts became more ‘fluent’ in this English-Uyghur hybrid, they reported gaining competence in reading the more direct representations of Uyghur as well (e.g., the original and IPA renderings), in the manner of Renduchintala et al. (2016). This additional knowledge, and the ability of the gloss to provide a second opinion about the meaning of a sentence, independent of the conclusions the SF or MT pipelines, was crucial in identifying systematic errors in SF classification (Sect. 6).

[11] This is thus not *lemmatization* per se—the lemma of all of these is *qatar*, with *-liq* being a suffix—but rather an attempt to find the most appropriate corresponding word in the lexicons, whether it is a lemma or not.

2.6 Date/time extraction

To support the SF coreference decision (Sect. 3.3) and time status classification (Sect. 5), we attempted to extract dates and other time-related information from incident language texts.

The basic approach is to refer to Unicode Common Locale Data Repository (CLDR),[12] a multilingual repository which includes various types of assets such as number patterns, date/time formats, etc. We converted the Uyghur date/time assets to a series of regular expressions and rules for date extraction, and attached the extracted date to the situation frame (using similar heuristics as used in Location matching in Sect. 4). This field was then used to help determine if two generated situation frames referred to the same event (events with incompatible date/time fields were not merged), and as one of the factors conditioning the decision between Current, Past Only, and Future Only.

After checkpoint 1, we incorporated additional useful patterns and expressions for extracting time information. For instance we added relative time expressions such as ‘N days ago’ or ‘last month’, and constructed an English-Uyghur time expression dictionary.

3 SF detection: the classifiers

3.1 Model I

Our first module identifies the presence in the text of a precompiled list of keywords and phrases that indicate the presence of SFs.

An initial English keyword list was obtained from the English corpora described in Sect. 2.3 by applying tf.idf directly; a similar initial key-phrase list was produced by combining pointwise mutual information (PMI) and tf.idf scores. We combined these lists and took the top 120 ranked words for each SF type, and manually refined and extended them to include additional morphological variants.

These keywords and phrases were then manually classified as *strong*, *weak*, or *misleading* for SF detection, partly based on tf.idf score and partly based on world knowledge and analysis of English text. Strong keywords tend, especially in concert, to reliably indicate the presence of an SF in the text. Weak keywords are not reliable indicators of specific needs or issues by themselves, but indicate that humanitarian needs and safety issues are likely. Weak keywords do not provide strong enough evidence for an SF type if too few strong keywords are present, and are never sufficient on their own for proposing the presence of an SF. Ambiguity also affects the strength of the keyword, with weak keywords having a higher degree of lexical ambiguity compared to strong ones, meaning that taken by themselves they may mislead the extractor by suggesting an SF when none is present, or by suggesting incorrect SFs.

[12] http://cldr.unicode.org.

As an example, the word *shelter* is a strong keyword for a shelter need, but the word *earthquake* is a weak keyword for a shelter need because not every earthquake results in a shelter need. *Nursing* is a weak keyword because it is ambiguous between mothers nursing children (not indicative of an SF) and nurses providing care to victims (indicative of an SF). *Power* and *patient* are generally misleading because they are ambiguous and too common in non-disaster situations. After misleading words were eliminated from the SF lists, we ultimately retained about 400 words/phrases all together.

To translate the English keywords to Uyghur we leveraged the Uyghur-English lexicons (Sects. 2.1, 2.4). Words that were not present in the lexicons were ignored. The resulting Uyghur keywords are used for assigning a binary score (yes or no) for each sentence for each SF type.

Using strong and weak keywords, there are two steps in assigning a per-sentence score. First, for SF identification, if any strong keywords appear in a sentence their associated SF is assumed to be present in that sentence. Second, given an SF of one type, we infer the presence of SFs of other types on the basis of their weak keywords, but only when the first (strong) type is known in general to co-ccur with the other (weak) types.

An additional complication arose when we discovered that many important meanings in Uyghur were present in the Uyghur-English dictionary only in the form of multi-word phrases. Many English verbs, for example, are expressed by a noun and light-verb pair in Uyghur. At Checkpoint 3, we extended our keyword system to handle multi-word Uyghur-English correspondences, rather than only single-word correspondences.

3.2 Model II

Our second model created a list of indicator words automatically. We trained a Naive Bayes classifier on the classified English texts (Sect. 2.3), initially using only word features, to classify single sentences into situation frame types. We then translated the model's words into Uyghur using the LDC-provided bilingual dictionary and inferred ones.

As a component in this model we also developed a graph-based word translation algorithm to extend the dictionary (Xu et al. 2016). We applied word2vec[13] (Mikolov et al. 2013a, b) to monolingual text (English and Uyghur separately) to obtain word vectors. Word similarity graphs were built for each language in which each node represents a word and each edge records the cosine similarity between the word embeddings of the two words. The similarity graphs of two languages were combined to induce the relation between the observed word translations (the seeded translations from the given bilingual dictionary) and the unobserved ones. Using the induced relation from word similarity graphs, the algorithm propagated the label information in the observed word translations to the unknown ones and produced new candidate translation lexicons.

[13] http://code.google.com/archive/p/word2vec/.

Table 5 Comparative Model I, Model II, and model intersection scores at Checkpoint I

Model	SFType			
	SFE	P	R	F1
Model I	1.832	0.272	**0.497**	**0.352**
Model II	1.765	0.221	0.304	0.256
Intersection	**1.082**	**0.424**	0.226	0.295

Bold values indicate our best performance in a primary submission (highest score for P, R, and F1, lowest score for SFE)

Initially this model introduced a very large number of words and other features and overgenerated SFs. For checkpoint 2, we performed feature selection on English to remove low-information features (as measured by PMI), reducing the number of features from 20 to 5 k and resulting in a 4% F1 score improvement on the English test set. Some English newswire documents were annotated internally by the ARIEL-CMU group (Sect. 2.3.4). We used the ones that contained no situation frames as negative examples for training. We also began using the more advanced lemmatizers (Sect. 2.5.2) for this checkpoint. We asked the native informant to validate the translations inferred by our graph-matching algorithm. Many of them turned out to be incorrect. Therefore we adjusted the threshold to include fewer inferred translations in our pipeline.

For checkpoint 3, we extended this model to include bi- and trigram features in order to capture multi-word Uyghur-English correspondences, for the reasons mentioned in Sect. 3.1.

3.3 Model combination

It was interesting to note that the two models produced rather different results. As is often the case, automated training was no match for human insight, and the manually tuned word/phrases lists of Model I outperformed the automatically acquired ones of Model II. After the tuning of Model II features and words described above its performance matched and later slightly exceeded that of Model I. Still, the results were far from identical.

We therefore experimented with different ways of combing their outputs. While Model I had the superior F1 scores, the official SFError metric is heavily weighted toward precision, and so it was far more important to produce *correct* though possibly too few SFs than to ensure coverage of *all* SFs. It was natural to simply intersect the outputs of the two models, and this became our top-scoring system entry.

Table 5 shows the comparative scores of Model I alone, Model II alone, and their intersection (our primary submission) at Checkpoint I. (We did not submit single-model systems beyond Checkpoint I.)

As each classifier model produced hypothesized SFs at the sentence level, intersection was also calculated at the sentence level. If both models found the same type of SF in the same sentence, and the SFs do not have incompatible location or time information, these SFs are merged; other SFs are deleted. Put another way, we only consider situation frames that can be found in a sentence in two different ways.

4 Location detection

We used a simple most-recent location heuristic to assign locations to each situation frame found: if the sentence that produced the situation frame contains a GPE or LOC named entity, we use that as the location. If the sentence contains more than one GPE or LOC named entities, we create new situation frames (with the same type and status values) for each additional named entity. On the other hand, if the sentence doesn't contain any location entities, we assign the last seen GPE or LOC named entity from the previous sentences or none (i.e. no Place mention) if no location entities could be found from the beginning of the document to the current position.

5 Status detection

Situation Frames (specifically those that describe 'needs' like Shelter rather than 'issues' like Widespread Crime) also have three 'Status' fields, intended to help mission planners determine whether the situation merits an immediate response:

- *Need* Does the need currently exist? Possible values: Current, Past Only, Future Only.
- *Relief* Is the need already being addressed, and is that relief sufficient? Possible values: Sufficient, Insufficient/Unknown Sufficiency, No relief known.
- *Urgency* Is a response urgently needed? Possible values: Urgent, Non-urgent.

The ARIEL-CMU system also achieved the best SFE scores in these categories; comparative P/R/F1 scores were not published.

The Need and Urgency categories, as specified in the SF annotation manual, have strong default responses (to Current and Non-urgent respectively); the Relief category does not have a clear default but the Sufficient label should (as described) be rare (and indeed was).

Accordingly, our best-performing Status-detection systems (in both the Mandarin dry-run evaluation and the Uyghur evaluation) were those that hewed closest to default values (Each of our Status-detection systems had a fallback to a default value when the system did not make a more specific decision, and given that this was usually the correct choice, our systems performed best when *not* making a decision).

Because of this, and because the Status score categories (SFType+Place+Need, SFType+Place+Relief, SFType+Place+Urgency) vary so heavily due to highly variant SFType and Place performance, the scores of these Status systems should probably not be interpreted as evidence for the superiority of a specific approach to Status.

Each of our SF Status systems started with hand-tuned English decision trees, based on spotting relevant keywords (e.g., 'now', 'sufficient', 'urgent') and taking into account the date/time fields produced by the system in Sect. 2.6, to classify sentences into appropriate Status categories or, if no such keywords are found, a special Default status that is later switched to the actual default status.

These decision trees were applied to Uyghur in one of three ways, paralleling the three transfer approaches described in Sect. 1.2.

1. Translating the Uyghur sentences to English (or more precisely, glossing them as in Sect. 2.5.4), then running the English decision trees on these.

Table 6 SFE, P, R, and F1 scores on SFType+Place+Need, SFType+Place+Relief, and SFType+Place+Urgency, for our primary systems at each checkpoint

Chkpt.	SFType+Place+Need			
	SFE	P	R	F1
CP1	**1.462**	0.084	0.047	0.060
CP2	1.758	0.060	0.052	0.055
CP3	1.767	**0.107**	**0.105**	**0.106**
Chkpt.	SFType+Place+Relief			
	SFE	P	R	F1
CP1	**1.486**	0.062	0.034	0.044
CP2	1.791	0.042	0.036	0.039
CP3	1.828	**0.073**	**0.070**	**0.071**
Chkpt.	SFType+Place+Urgency			
	SFE	P	R	F1
CP1	**1.458**	0.087	0.048	0.062
CP2	1.754	0.060	0.051	0.055
CP3	1.814	**0.078**	**0.076**	**0.077**

Bold values indicate our best performance in a primary submission (highest score for P, R, and F1, lowest score for SFE)

2. Using the Uyghur-English lexicons to translate each keyword, to make a Uyghur decision tree.
3. Using the English decision tree to label the English side of the parallel text, transferring the labels to the Uyghur sentence, and using those labeled Uyghur sentences to build an SVM classifier.

Our checkpoint 1 system used the third approach; our checkpoints 2 and 3 systems used the first approach; we made but did not end up submitting any systems using the second approach (Table 6).[14]

Which approach performed best depends on whether SFE or F1 is used as the evaluation metric, but as noted above, most of the variance in these scores comes from the performance of the SFType classification (Sect. 3) and Location linking (Sect. 4) in any case.

6 Error correction

6.1 Keyword validation

As noted above, one of the primary benefits of Model I (Sect. 3.1) is its scale: with only about 400 hand-tuned keywords, it more amenable to human inspection and error correction than the automated Model II classifier (Sect. 3.2).

[14] Compared to our SFType detection systems, the features in our English Status-detection decision trees focused comparatively more on functional words (e.g., words more often indicative of tense, aspect, or modality) than content words. We did not believe these words would translate well using a lexical feature-translation approach, so we did not submit any of these results as part of a primary submission.

When deriving translation lexicons from the LDC2016E57 Uyghur parallel data, mistranslations inevitably surface, as well as correct *word* translations that are not the correct sense for the incident domain. Since the Keyword model only begins with 400 English lexical entries, we could extract exactly those translation pairs and present them to the Native Informant (NI) for validation: "In a disaster context, is this a valid translation of this word?" This task is easy to understand and quick for the NI to execute, and provided what was probably the most significant per-hour knowledge gain during the evaluation.

6.2 Keyword-level error correction

Then, after running the entire pipeline on set0 (i.e., the LDC provided monolingual training text), we performed manual error correction on both the keywords and the resulting situation frames.

From each model, we took the most important keywords/features (i.e., those that had contributed to the identification of the greatest quantity of situation frames) and back-translated them into English (using the LDC2016E57 Uyghur-English lexicon) and Mandarin (using the online Uyghur-Mandarin dictionary referenced in the LDC2016E57 documentation directory). Inspection of the resulting English and Mandarin translations helped reveal which Uyghur keywords were likely to have non-disaster meanings and therefore lead to spurious situation frames.

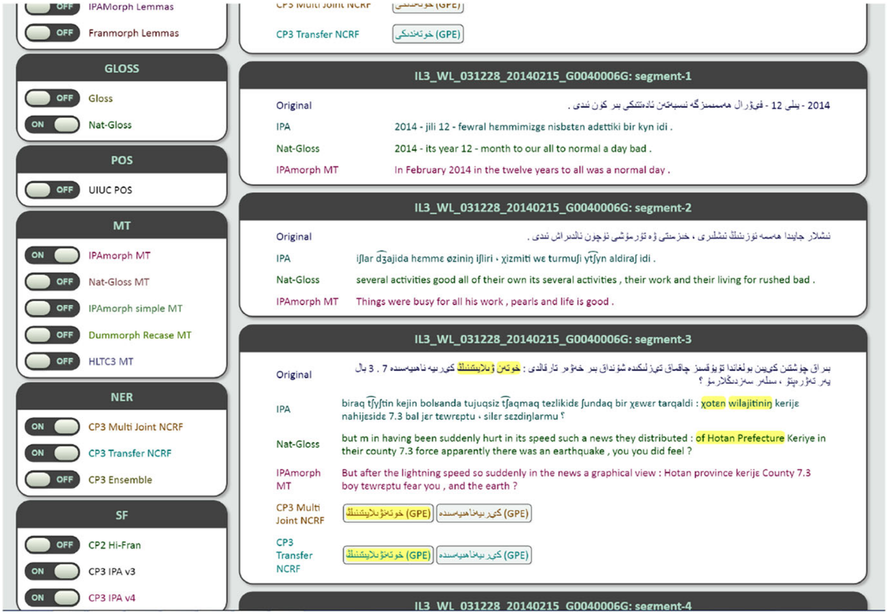
For example, the English keyword 'clean', which both models use as a feature to detect Water Supply needs (as in 'clean drinking water'), had a translation of مۇسۇلمان (*musulman*, 'Muslim') in one of our English-Uyghur lexicons (most likely in the sense of 'clean living' or 'halal'); this was subsequently removed from the keywords (and downweighted in the graph alignment model) to prevent the word مۇسۇلمان from leading to Water Supply situation frames.

Other keywords contained a correct back-translation for some senses of the word, but were judged to be too general to be a reliable indicator of a situation frame. For example, both models associated the keyword/feature 'search' with Search and Rescue needs, but one of this word's correspondents in the Uyghur-English lexicon was the verb lemma باق- (*baq*-). Back-translation revealed that this verb was extremely general in meaning (with back-translations including 'observe', 'take care of', 'help oneself to', and 'pertain to'), and it was likewise removed/downweighted as a Search and Rescue keyword.[15]

6.3 Sentence-level error correction

Finally, we examined a sample of set0 sentences that the models had identified as containing situation frames. Both the Native Informant and our non-speaker linguists attempted this task, the non-speaker linguists using an interface that simultaneously

[15] The error correction was performed on both models, but in the keyword model it was more straightforward to fix (i.e., by simply removing the keyword) and to know that the fix had worked.



OFF IPAMorph Lemmas

OFF Franmorph Lemmas

GLOSS

OFF Gloss

ON Nat-Gloss

POS

OFF UIUC POS

MT

ON IPAmorph MT

OFF Nat-Gloss MT

OFF IPAmorph simple MT

OFF Dummorph Recase MT

OFF HLTC3 MT

NER

ON CP3 Multi Joint NCRF

ON CP3 Transfer NCRF

OFF CP3 Ensemble

SF

OFF CP2 Hi-Fran

ON CP3 IPA v3

ON CP3 IPA v4

CP3 Multi Joint NCRF خوتەندىكى (GPE)

CP3 Transfer NCRF خوتەندىكى (GPE)

IL3_WL_031228_20140215_G0040006G: segment-1

Original 2014 - يىلى 12 - فېۋرال ھەممىمىزگە نىسبەتەن ئادەتتىكى بىر كۈن ئىدى .

IPA 2014 - jili 12 - fewral hemmimizge nisbeten adettiki bir kyn idi .

Nat-Gloss 2014 - its year 12 - month to our all to normal a day bad .

IPAmorph MT In February 2014 in the twelve years to all was a normal day .

IL3_WL_031228_20140215_G0040006G: segment-2

Original ئىشلار جايىدا ھەممە ئۆزىنىڭ ئىشلىرى ، خىزمىتى ۋە تۇرمۇشى ئۈچۈن ئالدىراش ئىدى .

IPA iʃlar d͡ʒajida hemme øziniŋ iʃliri ، χizmiti we turmuʃi yt͡ʃyn aldiraʃ idi .

Nat-Gloss several activities good all of their own its several activities , their work and their living for rushed bad .

IPAmorph MT Things were busy for all his work , pearls and life is good .

IL3_WL_031228_20140215_G0040006G: segment-3

Original بىراق چۈشتىن كېيىن بولغاندا تۇيۇقسىز چاقماق تېزلىكىدە شۇنداق بىر خەۋەر تارقالدى : خوتەن ۋىلايىتىنىڭ كېرىيە ناھىيەسىدە 7 . 3 بال يەر تەۋرەپتۇ ، سىلەر سەزدىڭلارمۇ ؟

IPA biraq t͡ʃyʃtin kejin bolʁanda tujuqsiz t͡ʃaqmaq tezlikide ʃundaq bir χewer tarqaldi : χoten wilajitiniŋ kerije nahijeside 7.3 bal jer tewreptu ، siler sezdiŋlarmu ؟

Nat-Gloss but m in having been suddenly hurt in its speed such a news they distributed : of Hotan Prefecture Keriye in their county 7.3 force apparently there was an earthquake , you you did feel ?

IPAmorph MT But after the lightning speed so suddenly in the news a graphical view : Hotan province kerije County 7.3 boy tewreptu fear you , and the earth ?

CP3 Multi Joint NCRF خوتەنۋىلايىتىنىڭ (GPE) كېرىيەناھىيەسىدە (GPE)

CP3 Transfer NCRF خوتەنۋىلايىتىنىڭ (GPE) كېرىيەناھىيەسىدە (GPE)

IL3_WL_031228_20140215_G0040006G: segment-4

Fig. 2 The ARIEL-CMU analyst interface, displaying a Uyghur news article in the original Perso-Arabic text, an IPA rendering, a naturalistic gloss, and machine translation output

showed the original text, an IPA rendering of it (Sect. 2.5.1), the ‘naturalistic glossing’ (Sect. 2.5.4), and our best machine translation output.

The analyst interface is seen in Fig. 2; the analyst can turn on and off different representations of the data on the left side, and on the right side, dynamic highlighting informs the analyst which pieces of each representation correspond to each other.

This task was much slower that keyword validation for the Native Informant, due in part to the number of spurious situation frames (after all, set0 was not specifically disaster-related text). On the other hand, the non-speaker linguist analysts could more quickly scan larger amounts of data to diagnose real problems in the model, but, obviously, could only spot some of the more egregious errors.

For example, user comments on disaster-related news articles were often some variation on “May God shelter us!”. Both systems understood this as a request for shelter, but nonetheless this is not a Shelter need SF. Since the word for ‘shelter’ is indeed a good Shelter-related keyword, we wrote a small ad-hoc classifier (using features like sentence length and presence of religious words) for the end of the pipeline that tries to classify ‘thoughts and prayers’-type comments from more informative comments.

It is worth emphasizing here that the non-speaker linguists were aided by information that was largely independent of the SF pipeline itself. Within the larger ARIEL-CMU system, the machine translation and SF pipelines diverge relatively early—they share only the G2P (Sect. 2.5.1) and lemmatization (Sect. 2.5.2) steps—so the naturalistic glossing and machine translation output provide a second (and third) opinion on what the data means. This is in contrast to an approach in which the data is first run through a MT system, and the model classifies the MT output. In that approach, using the MT output to diagnose classification errors would not count as a second opinion in the same way; both the classifier and the analyst would be using the same opinion (the MT system's opinion) of the meaning of the sentence.

7 Further research

Comparing the two SF classifiers' outputs, it is clear that their different approaches and data requirements make them complementary in a number of ways, and we plan to retain and develop each separately, as well as investigate methods for combining them. In challenges tasks with a lot of training data (even if not exactly parallel), the data-hungry Model II is likely to find more-obscure and situation-specific indicator words/features (for example, the names of relevant places) that Model I's fixed keyword list would obviously miss. But in data-poor situations, the converse holds, since Model II would not have enough material to learn reliable features. Balancing their relative contributions, and perhaps differentiating parts of the contributions depending on SF type or other aspects is a promising line of work.

We are also starting to look at including other information, obtained from background knowledge and compiled before the challenge task starts, that might be useful for need and SF type determination. We have developed a generic model of joint probabilities of various classes of information that might have some predictive effect, but populating this model before the incident is known with enough information to be relevant to each new incident remains a challenge.

8 Conclusion

The ARIEL-CMU situation frame detection pipeline took a different approach to English-Uyghur transfer than other teams' pipelines, translating the models' *features* into Uyghur rather than translating the Uyghur *text* into English. This approach led to the best scores (by both SFE and F1) among primary submissions in SFType classification, demonstrating the viability of a model translation approach for cross-linguistic information extraction. We believe this is in part because such systems depend on bilingual lexicons rather than bilingual text; in the absence of high-quality in-domain data, the former can be more easily supplemented than the latter. Parallel text can be utilized if it is available, but it is not a requirement of the system.

The core of the system is an intersection-based combination of two classifiers, one a manually curated keyword-spotting system and the other a Naïve Bayes classifier. This intersection (between a 'human-scale' system and a machine learning system) meant that constraining the former also constrained the latter, essentially allowing us to perform targeted human analysis and error correction that would otherwise be very difficult to perform on a pure machine learning system.

In the absence of Uyghur SF training data—indeed, in the absence of in-domain Uyghur data in general—this human analysis was crucial to avoid the inevitable miscategorizations that the system produces. However, this does not necessarily mean that the system cannot function without the presence of a native informant. We found that, given appropriate affordances in the user interface and access to multiple additional opinions about the meaning of the document, non-speaker analysts can rapidly gain enough proficiency to perform crucial data and error analysis.

This approach reflects the ‘omnivorous’ philosophy of the ARIEL-CMU project: making use of valuable but scarce resources (like parallel text, bilingual lexicons, annotations, and native informants) when they are available but able to fall back to more-readily-available but less-specific resources (like monolingual text, monolingual domain lexicons, and non-native speaker human judgments) when they are not. While there will always be some limitations on the former, we are exploring ways to significantly enhance the latter, by for example exploring the utility of geographical and numerical information typical of the SF needs in general. Being so data-poor, the LORELEI challenge encourages NLP to make some small but necessary steps toward using semantic knowledge and reasoning.

Acknowledgements This project was sponsored by the Defense Advanced Research Projects Agency (DARPA) Information Innovation Office (I2O), program: Low Resource Languages for Emergent Incidents (LORELEI), issued by DARPA/I2O under Contract No. HR0011-15-C-0114.

References

Baker M (1985) The mirror principle and morphosyntactic explanation. Linguistic Inquiry 16:373–415

Beesley KR, Karttunen L (2003) Finite state morphology. CSLI Publications, Stanford

Bharadwaj A, Mortensen D, Dyer C, Carbonell J (2016) Phonologically aware neural model for named entity recognition in low resource transfer settings. In: Proceedings of the 2016 conference on empirical methods in natural language processing. Association for Computational Linguistics, Austin, Texas, pp 1462–1472

Brown PE, Pietra VJD, Pietra SAD, Mercer RL (1993) The mathematics of statistical machine translation: parameter estimation. Comput Linguist 19(1):263–312

Dyer C, Chahuneau V, Smith NA (2013) A simple, fast, and effective reparameterization of IBM Model 2. In: Proceedings of the 2013 conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Atlanta, Georgia, pp 644–648

Forcada ML, Ginestí-Rosell M, Nordfalk J, O’Regan J, Ortiz-Rojas S, Pérez-Ortiz JA, Sánchez-Martínez F, Ramírez-Sánchez G, Tyers FM (2011) Apertium: a free/open-source platform for rule-based machine translation. Mach Transl 25(2):127–144

Frost R, Launchbury J (1989) Constructing natural language interpreters in a lazy functional language. Comput J 32:108–121

Hutton G (1992) Higher-order functions for parsing. J Funct Progr 2:323–343

Hutton G, Meijer E (1988) Monadic parser combinators. J Funct Progr 8:437–444

Lewis MP, Simons GF, Fennig CD (2015) Ethnologue: languages of the world, 18th edn. SIL International, Dallas, Texas

Linden K, Silfverberg M, Axelson E, Hardwick S, Pirinen T (2011) HFST-framework for compiling and applying morphologies. Commun Comput Inf Sci 100:67–85

Mikolov T, Chen K, Corrado G, Dean J (2013a) Efficient estimation of word representations in vector space. CoRR abs/1301.3781

Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013b) Distributed representations of words and phrases and their compositionality. In: Burges CJC, Bottou L, Welling M, Ghahramani Z, Weinberger KQ (eds) Advances in neural information processing systems, vol 26. Curran Associates, Inc., pp 3111–3119

Olteanu A, Castillo C, Diaz F, Vieweg S (2014) Crisislex: a lexicon for collecting and filtering microblogged communications in crises. In: Proceedings of the AAAI conference on weblogs and social media (ICWSM’14), Ann Arbor, MI, USA

Renduchintala A, Knowles R, Koehn P, Eisner J (2016) Creating interactive macaronic interfaces for language learning. In: Proceedings of ACL-2016 System Demonstrations, Association for Computational Linguistics, Berlin, Germany, pp 133–138

Stenetorp P, Pyysalo S, Topić G, Ohta T, Ananiadou S, Tsujii J (2012) brat: a web-based tool for NLP-assisted text annotation. In: Proceedings of the demonstrations at the 13th conference of the European chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Avignon, France, pp 102–107

Strassel S, Tracey J (2014) LORELEI language packs: data, tools, and resources for technology development in low resource languages. In: LREC 2016: 10th edition of the language resources and evaluation conference, Portoroz, pp 3273–3280

Strassel S, Bies A, Tracey J (2017) Situational awareness for low resource languages: the LORELEI situation frame annotation task. In: SMERP2017: first international workshop on exploitation of social media for emergency relief and preparedness, Aberdeen

Temnikova I, Castillo C, Vieweg S (2015) Emterms 1.0: a terminological resource for crisis tweets. In: Proceedings of the international conference on information systems for crisis response and management (ISCRAM'15), Kristiansand, Norway

Washington JN, Ipasov IS, Tyers FM (2014) Finite-state morphological transducers for three Kypchak languages. In: Proceedings of the 9th conference on language resources and evaluation, LREC2014

Xu R, Yang Y, Liu H, Hsi A (2016) Cross-lingual text classification via model translation with limited dictionaries. In: Proceedings of the 25th ACM international on conference on information and knowledge management, ACM, pp 95–104