Query by Example for Cross-Lingual Event Retrieval

Sheikh Muhammad Sarwar
CIIR, College of Information and Computer Sciences
University of Massachusetts Amherst
smsarwar@cs.umass.edu

James Allan
CIIR, College of Information and Computer Sciences
University of Massachusetts Amherst
allan@cs.umass.edu

ABSTRACT

We propose a Query by Example (QBE) setting for cross-lingual event retrieval. In this setting, a user describes a query event using example sentences in one language, and a retrieval system returns a ranked list of sentences that describe the query event, but from a corpus in a different language. One challenge in this setting is that a sentence may mention more than one event. Hence, matching the query sentence with document sentence results in a *noisy matching*. We propose a Semantic Role Labeling (SRL) based approach to identify event spans in sentences and use a state-of-the-art sentence matching model, Sentence BERT (SBERT) to match event spans in queries and documents without any supervision. To evaluate our approach we construct an event retrieval dataset from ACE [20] which is an existing event detection dataset. Experimental results show that it is valuable to predict event spans in queries and documents and our proposed unsupervised approach achieves superior performance compared to Query Likelihood (QL), Relevance Model 3 (RM3) and SBERT.

KEYWORDS

Cross-lingual IR, Event Retrieval, Sentence BERT

ACM Reference Format:
Sheikh Muhammad Sarwar and James Allan. 2019. Query by Example for Cross-Lingual Event Retrieval. In *SIGIR '20: ACM SIGIR Conference on Research and Development in Information Retrieval , July 25–30, 2020, Xi'an, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Query by Example (QBE) is an effective alternative to keyword queries for identifying user information need. It has been applied to retrieve entities and documents from unstructured text corpora [16, 17, 19], entities from knowledge graphs [9], and tuples from relational databases [5]. QBE approaches are motivated by the fact that it is often easier for a user to express an information need with examples rather than a natural language description. Consider the case where a user wants to find all the *jail release* events from a corpus. To start this process, she retrieves a few documents with combination of keywords such as *jail*, *release*, *sentence*, etc., and finds sentences from those documents that mention a jail release event. Although these sentences constitute a representation of her information need (query), traditional retrieval approaches do not provide support for such an event query. A sentence matching model that computes similarity between a pair of sentences can be a remedy to this problem. However, our experiments suggest that performance of a state-of-the-art unsupervised sentence matching model is sub-optimal for event matching.

We study the above mentioned event matching problem in a cross-lingual setting - i.e., we assume that the language of example sentences and corpus sentences are different. Although Cross-Lingual Information Retrieval (CLIR) is a well-studied problem, most CLIR studies are targeted towards document retrieval [18]. To the best of our knowledge, there is no study or available testbed for studying CLIR or even mono-lingual IR for example-driven event retrieval. Such a setting would be very useful for journalists, security agency personnel, and political scientists. This motivated us to create a testbed and evaluate standard retrieval approaches for our task, Cross-Lingual Event Retrieval with Query by Examples (CLER-QBE).

To solve CLER-QBE, we follow a popular CLIR approach that uses two stages: query translation and retrieval [12]. We translate example sentences that constitute our event query using a commercial Machine Translation (MT) system and focus on the retrieval problem. It is challenging to retrieve sentences containing a target event with translations of examples sentences for two reasons: i) translated example sentences are noisy because of MT error; ii) only a sub-sequence of tokens in the translated example sentences describes the target event that holds for corpus sentences too. Both these issues make it challenging to understand user intent and match event mentions in translated examples and corpus sentences. They result in a phenomenon we refer as *noisy matching*.

To alleviate the effect of the noisy matching problem, we assume to have event trigger annotation for our example sentences. Consider the sentence describing a jail release event: "Pasko, whose sentence included time served, was **released** in January for good behavior after serving more than two-thirds of the sentence". Note mention of three events: sentence, jail release, and sentence serving completion. We assume that a user interested in the jail release event would provide us with the trigger keyword *released* along with the example sentence so that we can extract appropriate context around the trigger to understand the user intent. This is still problematic from the perspective of retrieval because even if we are able to extract the appropriate query, we do not know what span of the document we should match with the query context, as documents could also contain more than one event.

To extract event extents from documents and match them with query context we propose to use PredPatt, an unsupervised technique for Semantic Role Labeling (SRL) [22]. PredPatt identifies the predicates and their corresponding arguments from a sentence. We use that information to predict event spans in documents. Once the document event spans are identified, we match them with query context using a recently proposed Sentence-BERT (SBERT) model [14]. The original BERT model does not provide effective out-of-the-box sentence embeddings without fine-tuning [14]. SBERT is fine-tuned with Natual Language Inference (NLI) data and it is able

SIGIR '20, July 25–30, 2020, Xi'an, China
2019. ACM ISBN 978-1-4503-9999-9/18/06...$15.00
https://doi.org/10.1145/1122445.1122456

to create sentence embeddings that significantly outperforms other state-of-the-art models on semantic textual similarity tasks. Finally, to describe our contributions concisely, we propose the task of CLER-QBE, construct a standard testbed, evaluate classical retrieval approaches on that, and propose an effective SRL-based technique to predict document event spans as well as an unsupervised matching model to match query context with the predicted spans.

2 PROBLEM FORMULATION

$Q_e = \{s^1_{src}, s^2_{src} \ldots, s^n_{src}\}$ is an event query that consists of n example sentences mentioning a target event, $e = \{s^1, s^2, \ldots, s^n\}$ in *src* language. For example, $Q_{\text{jail release}} = \{s^1_{Arabic}\}$ indicates that a user has provided an example sentence describing a *jail release* event in Arabic and wants to retrieve sentences describing *jail release* events in another language. Q_e is issued against a corpus, $D_{trg} = \{d^1_{trg}, d^2_{trg}, \ldots, d^m_{trg}\}$ of m sentences written in *trg* language. There is a relation, $Event(d^i_{trg}) \subset E = \{e_1, e_2, \ldots e_l\}$ that maps a sentence d^i_{trg} to a set of events, E. We assume query event $e \in E$ for the sake of evaluation. $Event(d^i_{trg}) = \{\phi\}$ indicates that d^i_{trg} does not mention any event. The task is to retrieve a ranked list $R = (d^1_{trg}, d^2_{trg}, \ldots, d^k_{trg})$ of k sentences mentioning e. A sentence d^i_{trg} in the ranked list is relevant if $e \subseteq Event(d^i_{trg})$; otherwise it is non-relevant.

Our problem setting assumes that the user has annotated example sentences with event *triggers* or *nuggets*. This assumption is based on event detection literature where an event mention contains a main word or phrase that evokes the event [7, 13]. To illustrate this we provide an example from our dataset: "Pasko, whose **sentence** included time served, was **released** in January for good behavior after **serving** more than two-thirds of the sentence". This example actually describes three events: i) *Pasko* was sentenced, ii) he was released from jail, and iii) he served in a jail. If the user annotates the example sentence with the keyword *released* it probably means that she is looking for jail release events. As we have user annotated triggers, our query description is further enriched as $Q_e = \{(s^1_{src}, t^1_{src}),\ (s^2_{src}, t^2_{src}),\ \ldots,\ (s^n_{src}, t^n_{src})\}$. Our query is a set of 2-tuples where the second element of the tuple denotes the event trigger. We use $Q_e = \{s^1_{src}, s^2_{src}, \ldots, s^n_{src}\}$ and $Q^t_e = \{t^1_{src}, t^2_{src}, \ldots, t^n_{src}\}$ as sentence query and trigger query, respectively. Sentence and trigger queries based on the above example would be $Q_{\text{jail release}} = \{\text{Pasko, whose ... released ... sentence.}\}$ and $Q^t_{\text{jail release}} = \{\text{released}\}$.

3 APPROACH

Our approach consists of four components: *Query Translation*, *Document Scoring*, *Matching Model* and *Event Span Detection*.

Query Translation. One common practice in cross-lingual information retrieval is to translate a search query using an off-the-shelf MT model, and perform mono-lingual retrieval using the translated query [12]. We take the same approach - i.e., we translate Q_e and Q^t_e into target language using a commercial MT model to obtain $\widetilde{Q}_e = \{\tilde{s}^1_e, \tilde{s}^2_e, \ldots, \tilde{s}^n_e\}$ and $\widetilde{Q}^t_e = \{\tilde{t}^1_e, \tilde{t}^2_e, \ldots, \tilde{t}^n_e\}$, respectively.

Document Scoring. Now that our sentence and trigger queries are translated into the target language, we use a mono-lingual sentence matching model, M_s to compute similarity between our queries and documents. Given M_s, a sentence matching model we compute the score of a document in the target language as, $score(d^i_{trg}) = \sum_{\tilde{s}^j_e \in \widetilde{Q}_e} M_s(\tilde{s}^j_e, d^i_{trg})$. Similarly, we use a model M_t to match triggers with corpus sentences and compute similarity scores using $score(d^i_{trg}) = \sum_{\tilde{t}^j_e \in \widetilde{Q}^t_e} M_t(\tilde{t}^j_e, d^i_{trg})$. Sorting the documents using the scores computed by each model results in two ranked lists that we combine using the reciprocal rank fusion approach [3]. The intuition behind combining lists is that they capture different aspects of matching. The trigger matching model does not include context while the sentence matching model includes it. We provide discussion and justification for using the ranked list fusion approach in the experimental results section.

Matching Model. Our trigger matching model, M_t, is query likelihood approach. As triggers do not contain any contextual information, unigram statistics are sufficient to establish matching. As sentence matching model, M_s, we use a very recent architecture, Sentence BERT (SBERT) proposed by Reimers and Gurevych [14]. SBERT adds a pooling operation to the output of BERT to derive a fixed sized sentence embedding. Similar to the authors we use the mean pooling strategy to compute a fixed size representation for sentences. With a fixed size representation of a pair of sentences we use cosine similarity to compute the similarity between them. However, one problem with event retrieval is a sentence usually mentions more than one event, which holds for both query and document sentences in our setting. To match the query event with the document event accurately we focus on the relevant part of the example sentence and the corpus sentence. The next section describes how we find these relevant parts.

Event Span Detection. Given $\widetilde{Q}_e$ we compute matching scores of each $\tilde{s}^j_e \in \widetilde{Q}_e$ with each $d^i_{trg} \in D_{trg}$ using M_s. Before doing that we need to consider that a target event e is usually mentioned by a subsequence of tokens in the example sentence $\tilde{s}^j_e$. Considering the entire sentence as the search intent would result in noisy matching. To alleviate this problem we locate the trigger $\tilde{t}^j_e$ in $\tilde{s}^j_e$ and take a window of information around $\tilde{t}^j_e$. As $\tilde{t}^j_e$ and $\tilde{s}^j_e$ are translations of t^j_e and s^j_e, sometimes $\tilde{t}^j_e$ cannot be located in $\tilde{s}^j_e$ even if t^j_e appears in s^j_e. In that case we compute word embedding similarity of $\tilde{t}^j_e$ and all others tokens in $\tilde{s}^j_e$ and select the location of the highest scored token. Assuming the location is l, we consider a token span starting from $l - w$ to $l + w$ to capture a window w of tokens around the translated event trigger. We refer to this token span as query context. This approach also needs to be applied to documents as they may also mention more than one event.

In order to find event spans in a document we use a Semantic Role Labeling Approach (SRL) to find predicate argument structure from a sentence. Given a sentence SRL is used to answer basic questions about sentence meaning, including "who" did "what" to "whom," etc [2]. We use an unsupervised SRL approach Predictive Patterns (PredPatt) [21] to find predicate and arguments and use those to predict event spans from documents. PredPatt is lightweight, fast, and unlike other supervised SRL approaches, it does not need to adapt to a target domain with further training [6, 22]. It uses a set of non-lexicalized, extensible and interpretable patterns on the Universal Dependency (UD) [4] parse of a sentence to extract predicates and arguments. An important reason to select PrePatt is it works over Universal Dependency (UD) parse that enables it to extract predicate and arguments in almost any language.

To illustrate how we use PredPatt to predict event spans, consider the example provided in our problem definition section: "Pasko, whose **sentence** included time served, was **released** in January for good behavior after serving more than two-thirds of the sentence". The predicates and their corresponding arguments found by running PredPatt on the example are shown in Table 1. We predict event spans by considering the minimum size token window that covers a predicate and all its arguments. As a result, a document d^i_{trg} is decomposed into f token spans i.e. $d^i_{trg} = \{d^{i1}_{trg}, d^{i2}_{trg}, \ldots, d^{if}_{trg}\}$. In order to compute the score of d^i_{trg} with respect to example sentence $\tilde{s}^j_e$ we take $max_{1,2,\ldots f} M_s(\tilde{s}^j_{e,trg}, d^{if}_{trg})$ - i.e., we take the maximum of the scores of the token spans.

Table 1: Event Span Prediction Using PredPatt [22]

Predicate	Arguments	Predicted Event Spans
included	{sentence, time}	sentence **included** time
released	{Pasko}	Pasko , whose sentence included time served , was **released**
serving	{two-thirds}	**serving** more than two-thirds

4 EXPERIMENTAL SETUP AND RESULTS

Dataset Construction. We adopt the ACE 2005 multilingual event detection dataset provided by the Linguistic Data Consortium (LDC) [20] to evaluate CLER-QBE. ACE 2005 provides sentences in *English*, *Arabic*, and *Chinese* with event types annotated by human judges. There are different number of sentences in different languages and the number of event types also vary. We pre-processed the original ACE 2005 dataset[1] and then performed an analysis of the dataset based on event types. We report a few frequent event types along with the number of sentences that mentions those types in Table 2.

In our processed version of ACE, each sentence is POS tagged, annotated with golden (truth) event type with event trigger span indicated, and annotated with golden entity type with entity token span indicated. We used the Stanford CoreNLP English, Arabic and Chinese libraries [8] for preprocessing. Our processed version of ACE contains 16249, 1458, and 2088 sentences in English, Chinese, and Arabic, respectively. Among them 5224, 487, and 2059 sentences mention at least one event. As English has the largest number of sentences, we construct our retrieval corpus from English. Each sentence in this retrieval corpus is relevant to a specific type of event or does not indicate an event at all. We assume each event type as a query, randomly draw Arabic and Chinese example sentences for that event type, and retrieve sentences from the English corpus to perform evaluation.

Experimental Setting. We use *Indri* search framework to index our English corpus and create relevance judgments based on ground truth event annotations. We use existing implementations of PredPatt [2] and SBERT [3]. We use TrecTools [4] to evaluate our retrieval runs and perform reciprocal rank fusion. We use window size of five around the trigger words in example sentences to determine query context. Our adopted ACE dataset and source codes to generate all the experimental results are available [5].

[1] https://github.com/nlpcl-lab/ace2005-preprocessing

Table 2: Highly occurring events in ACE with the number of sentences describing them in different languages

Event Type	English	Chinese	Arabic
Movement:Transport	713	99	392
Conflict:Attack	1510	74	455
Contact:Meet	280	44	190
Transaction:Transfer-Money	187	24	42
Life:Die	584	34	213

Experimental Results. We report retrieval performance in terms of Precision@10 and Mean Average Precision (MAP) on the ACE English retrieval corpus using Chinese and Arabic queries containing different number of example sentences. We use three retrieval approaches: QL (Query Likelihood), RM3 (Relevance Model 3) and Sentence BERT (SBERT) [14] and three different example query types: sentences (S), triggers (T), combined (ST). The process of constructing a combined (ST) query is illustrated in section 3 and we use it with SBERT matching model. As our proposed query construction method includes an SRL component, we refer to this approach as SBERT-ST (SRL). Thus we have five baseline approaches: QL-T (QL with Trigger Query), QL-S (QL with Sentence Query), RM3-T, RM3-S, SBERT-S, along with two proposed approaches SBERT-ST (SRL) and SBERT-ST (SRL + Fusion). SBERT-ST (SRL + Fusion) is the reciprocal rank fusion of QL-T and SBERT-ST (SRL). Note that QL-S and RM3-S do not directly support sentence queries. Hence, we construct a bag-of-words query from the example sentences by extracting unique terms from them. All the Chinese and Arabic sentences as well as trigger queries were translated by Google Machine Translation API [6].

Figure 1 reports the precision@10 and Mean Average Precision (MAP) for retrieval with Chinese and Arabic Queries with increasing number of examples. One important thing to note that trigger queries (QL-T, RM3-T) result in much better performance than paragraph queries (QL-S, RM3-S). It happens because we have a small retrieval corpus and we do not lose precision by matching ambiguous triggers. For example, there is less chance of matching a *sports attack* event than a *military attack* event with keyword *attack* as a query. The failure of the baseline sentence query approaches (QL-S, RM3-S, SBERT-S) is explainable by the noisy matching phenomenon that happens when the entire example and document are considered for matching. Our proposed approach SBERT-ST (SRL) outperforms all the baseline approaches with sentence queries in terms of Precision@10 for any number of examples. We observe gain in MAP for Arabic queries, while for Chinese queries this gain is achieved with more than four examples. Finally, to combine the strength of trigger and paragraph queries, our proposal SBERT-ST

[2] https://github.com/hltcoe/PredPatt
[3] https://github.com/UKPLab/sentence-transformers
[4] https://github.com/joaopalotti/trectools
[5] URL provided upon publication
[6] https://cloud.google.com/translate

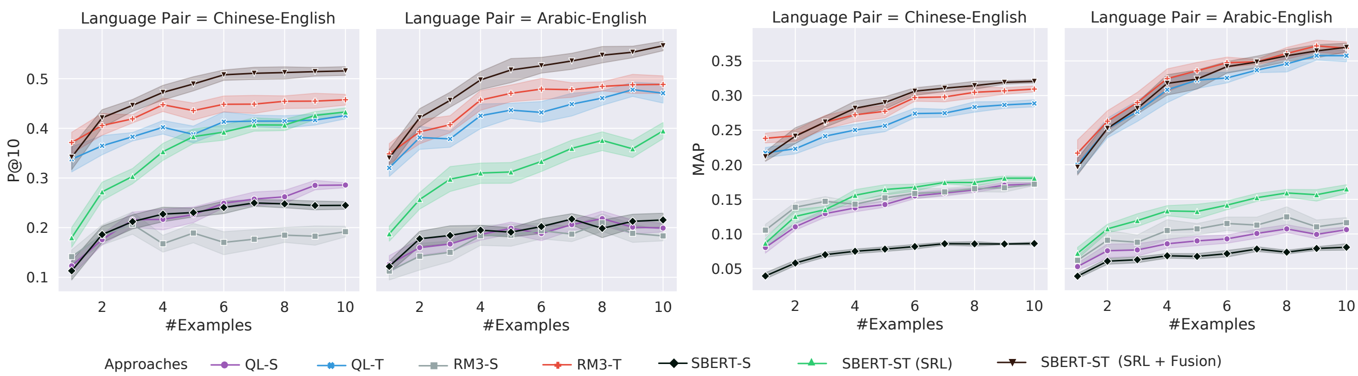

Figure 1: Retrieval performance in terms of Precision@10 and MAP for two language pairs with increasing number of examples. We randomly sample ten sets of k-examples query and plotted the mean with 95% confidence interval.

(SRL + Fusion), which is a reciprocal rank fusion of QL-T and SBERT-S (SRL), outperforms all the baselines in terms of P@10. Improvement in MAP is also observed but not for Arabic queries.

5 RELATED WORK

Event detection from unstructured text is a closely related task where an event mention in text is classified into a set of predefined event types. For example, given the sentence "A police officer killed a civilian in New Jersey today", an event detection system identifies the word "killed" as a trigger for the event "Death". Event detection task is generally solved using supervised machine learning approaches with fixed number of event classes [11]. Event detection from social media streams is a slightly different task which is solved using classification and summarization. For example, Alsaedi et al. [1] used a classification approach to detect *disruptive events* from social media and the summarized contents of such events to show sub-events of interest to users. In our setting, users will be able select examples from such summarized contents and use them as queries. This makes event detection orthogonal to what we want to achieve.

Metzler et al. [10] proposed *microblog event retrieval* task and used keyword queries to perform retrieval on Twitter corpus constructed over a period of time. Their approach involved detection of time-spans in which a target event occurred and summarization of the contents in that time-span for describing the event. Rudra et al. [15] explored a similar approach retrieve disaster related information e.g., about infrastructure damage, urgent needs of affected people. They identified sub-events using noun-verb pairs that closely occur in different tweets, for example "airport shutdown". Finally, they summarized the contents associated with the sub-events using an Integer Linear Programming approach. These approaches are fundamentally focused towards single keyword or phrase queries such as *earthquake* to detect events from Microblogs. Our queries are constructed from example event descriptions, and they are in different language from corpus language.

6 CONCLUSION

We proposed the CLER-QBE task and took a first step to evaluate it using an existing event detection dataset. We explored classical information retrieval approaches as well as state-of-the-art sentence embedding approaches to solve this task. We found that event triggers as examples are much more effective queries than example sentences. However, success of our approach in predicting event spans in examples and corpus sentences indicate that there is value in combining information from triggers and sentences. In future, we plan to extend the retrieval corpus in this dataset with ambiguous triggers so that leveraging event context from example sentences becomes more useful.

ACKNOWLEDGMENTS

This research is based upon work supported in part by the Center for Intelligent Information Retrieval, and in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via Contract No. 2019-19051600007 under Univ. of Southern California subcontract no. 124338456. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

REFERENCES

[1] Nasser Alsaedi, Pete Burnap, and Omer Rana. 2017. Can We Predict a Riot? Disruptive Event Detection Using Twitter. *ACM Trans. Internet Technol.* (2017), 26.
[2] Xavier Carreras and Lluís Màrquez. 2005. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *CoNLL '05*.
[3] Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. 2009. Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods. In *SIGIR '09*.
[4] Marie-Catherine de Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal Stanford dependencies: A cross-linguistic typology. In *LREC '14*.
[5] Anna Fariha, Sheikh Muhammad Sarwar, and Alexandra Meliou. 2018. SQuID: Semantic Similarity-Aware Query Intent Discovery. In *SIGMOD '18*.
[6] Silvana Hartmann, Ilia Kuznetsov, Teresa Martin, and Iryna Gurevych. 2017. Out-of-domain FrameNet Semantic Role Labeling. In *EACL '17*.
[7] Viet Dac Lai and Thien Nguyen. 2019. Extending Event Detection to New Types with Learning from Keywords. In *W-NUT '19*.
[8] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Prismatic Inc, Steven J. Bethard, and David Mcclosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *ACL '14*.
[9] Steffen Metzger, Ralf Schenkel, and Marcin Sydow. 2017. QBEES: query-by-example entity search in semantic knowledge graphs based on maximal aspects,

diversity-awareness and relaxation. *Journal of Intelligent Information Systems* (2017), 333–366.
[10] Donald Metzler, Congxing Cai, and Eduard Hovy. 2012. Structured Event Retrieval over Microblog Archives. In *NAACL '12*.
[11] Thien Huu "Nguyen and Ralph" Grishman. 2015. "Event Detection and Domain Adaptation with Convolutional Neural Networks". In *"ACL '15"*.
[12] Jian-Yun Nie. 2010. *Cross-Language Information Retrieval*. Morgan & Claypool Publishers.
[13] Nils Reimers and Iryna Gurevych. [n. d.]. Event nugget detection, classification and coreference resolution using deep neural networks and gradient boosted decision trees. *Transfer* ([n. d.]), 554.
[14] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP-IJCNLP '19*.
[15] Koustav Rudra, Pawan Goyal, Niloy Ganguly, Prasenjit Mitra, and Muhammad Imran. 2018. Identifying Sub-Events and Summarizing Disaster-Related Information from Microblogs. In *SIGIR '18*.
[16] S.M. Sarwar, John Foley, Liu Yang, and James Allan. 2019. Sentence Retrieval for Entity List Extraction with a Seed, Context, and Topic. In *ICTIR '19*.
[17] Sheikh Muhammad Sarwar and James Allan. 2019. SearchIE: A Retrieval Approach for Information Extraction. In *ICTIR '19*.
[18] Sheikh Muhammad Sarwar, Hamed Bonab, and James Allan. 2019. A Multi-Task Architecture on Relevance-based Neural Query Translation. In *ACL '19*.
[19] Mark D. Smucker and James Allan. 2006. Find-Similar: Similarity Browsing as a Search Tool. In *SIGIR '06*.
[20] Christopher et al. Walker. 2006. ACE 2005 Multilingual Training Corpus.
[21] Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. Universal Decompositional Semantics on Universal Dependencies. In *EMNLP '16*.
[22] Sheng Zhang, Rachel Rudinger, and Benjamin Van Durme. 2017. An Evaluation of PredPatt and Open IE via Stage 1 Semantic Role Labeling. In *IWCS 2017*.