

# Knowledge-based Approach for Event Extraction from Arabic Tweets

Mohammad AL-Smadi  
Computer Science Department  
Jordan University of Science and Technology  
P.O.Box: 3030 Irbid 22110, Jordan

Omar Qawasmeh  
Computer Science Department  
Jordan University of Science and Technology  
P.O.Box: 3030 Irbid 22110, Jordan

**Abstract**—Tweets provide a continuous update on current events. However, Tweets are short, personalized and noisy, thus raises more challenges for event extraction and representation. Extracting events out of Arabic tweets is a new research domain where few examples – if any – of previous work can be found. This paper describes a knowledge-based approach for fostering event extraction out of Arabic tweets. The approach uses an unsupervised rule-based technique for event extraction and provides a named entity disambiguation of event related entities (i.e. person, organization, and location). Extracted events and their related entities are populated to the event knowledge base where tagged tweets' entities are linked to their corresponding entities represented in the knowledge base. Proposed approach was evaluated on a dataset of 1K Arabic tweets covering different types of events (i.e. instant events and interval events). Results show that the approach has an accuracy of, 75.9% for event trigger extraction, 87.5% for event time extraction, and 97.7% for event type identification.

**Keywords**—Event Extraction; Knowledge base; Entity linking; Named entity disambiguation; Arabic NLP.

## I. INTRODUCTION

Social media sites such as Facebook and Twitter provide the most updated events leveraging the social generated content. Hundreds of millions of tweets are provided every day covering a variety of events and news. However, extracting structured information about events from these tweets holds a great promise especially when it comes to visualize events in more appealing way according to users' interests. Nevertheless, linking entity mentions in tweets with their events to their corresponding entities in the knowledge base fosters many research fields such as knowledge base population, questions answering, and information integration.

Many of previous research on event extraction [1]–[4] have focused on document level extraction such as News articles and Blogs, whereas few examples can be found on event extraction from noisy text such as tweets [5]–[10]. However, research targeting event extraction out of Arabic text is limited [11]–[13] and to the best of our knowledge there is only one concurrent research reported on event extraction out of Arabic tweets [14].

In general, extracting information from noisy text such

as social media posts is challenging. Such posts are disorganized and require automated approaches of information extraction and categorizing. For instance, tweets are short and self-contained which make them lack useful discourse information such as contextual information. According to [10], Twitter holds a set of challenges when it comes to event extractions such as: (a) tweets are personalized and mainly hold information about owner daily activities that of interest for their close social network only. (b) Tweets are short and self-contained and usually lack information about their context which causes NLP tools to perform poorly. On the other hand, such challenges hold great promises to enhance and adapt state-of-the-art NLP tools accordingly. (c) Twitter users informally contribute to a variety of topics and domains thus complex to categorize.

With the advances of Semantic Web and the so-called Web 3.0 folksonomy-based social environments, interoperability of knowledge management is a key challenge where semantics play an important role in facing it [15]. However, this cannot be achieved without bridging Web data with knowledge bases through linking named entity mentions appearing in Web material with their corresponding entities in a knowledge base [16].

Entity linking plays a major role when it comes to populate information to the knowledge base, or integrating extracted information from the Web. Adding newly extracted information to the knowledge base requires an entity linking step between entity mentions (in the text) and their corresponding entities in the knowledge base [17], [18]. However, non of the events extraction related work has focused on the events entities disambiguation or linking.

In this research, an unsupervised approach for event extraction out of Arabic tweets is discussed. The approach tags the event expression and the related entities and link them to the knowledge base entities and events. To the best of our knowledge there is no research that links events entity mentions to the Linked Open Data (LOD) as part of the event extraction process. This research links events' entity mentions (i.e. Person, Location, and Organization) to their corresponding entities in Wikipedia or DBpedia. This process is handled through an ontology based knowledge base that has been designed to represent event entities and link them to LOD. Moreover, newly extracted events (not available in the knowledge base) are

populated to the knowledge base. Events represented in the knowledge base are used to provide services such as a calendar, or a time-line of events.

The rest of this paper is organized as follows: Section 2 sheds the light on previous and related work for event extraction out of Arabic text in general and Arabic tweets in particular. Section 3 discusses the proposed approach and explains the method of event extraction and representation. Section 4 reports the approach evaluation procedure and results. Section 5 discusses the evaluation results. Finally, Section 6 concludes this work and provides plans for future work.

## II. RELATED WORK

Recently, the problem of event extraction in general and event extraction from noisy text in particular has gained the researchers interest. However, few examples can be found on Arabic tweets. Related work for event extraction can be categorized according to the used corpora into: (a) document-level event extraction, and (b) sentence-level event extraction.

### A. Document-Level Event Extraction

Allan et al. [1] propose an approach that is able to extract events out of news articles. The approach employs a feature extraction algorithm that processes each news article sequentially to build a query representation for each one. Then uses a comparison algorithm to determine the article that contains events and add it to their database. The approach was tested on 15,863 news articles from the period of July-1994 to June-1995. Results showed that the approach was able to detect events with F-measure = 0.49.

Ahn [19] proposes an approach that uses a number of machine learning techniques in order to extract events from ACE corpus. The process of extracting the events was split into a four main sub tasks: (a) Anchor identification, (b) Arguments identification, (c) Attribute assignment and (d) Event co-reference. The approach was trained using a set of features such as: lexical features, WordNet features and Dependency features. Results achieved was with F-measure = 0.601 for the overall subtasks.

Chambers and Jurafsky [4] propose an unsupervised approach that learns and extracts a template scheme structure automatically from text and produces a set of linked events such as war events. The approach was evaluated using the MUC-4 terrorism dataset [20]. Results showed that the approach was able to extract template structure very similar to the annotated gold structures (F-measure = 0.40).

On the other hand, some examples can be reported for event extraction out of Arabic documents. For instance, Abuleil [11] uses a lexicon-based approach to extract event mentions out of Arabic articles. Out of 300 articles the approach was able to detect 439 events out of 467 events.

Saleh et al. [12] propose a machine learning method to extract Arabic temporal and numerical events. They created an Arabic Treebank and used it to tag temporal expressions in Arabic text.

Aliane et al. [13] use an unsupervised approach for text segmentation and a rule-based approach to extract events' expressions and their locations. The approach was tested on a corpus of 30 articles crawled from the web, and was able to extract 168 verbal events out of 268.

### B. Sentence-level Event Extraction

Lin et al. [5] propose an approach that deals with popular event tracking (PET) in online websites by focusing on the interaction between textual structure and social networks. A statistical approach that models the popularity of events over time period was evaluated on two different datasets (i.e., DBLP and twitter) and showed a good enhancement over similar approaches that used the same datasets.

Ritter et al. [10] present TwiCal for extracting events out of tweets. For a given tweet, their system was able to extract named entities with event phrases, events dates and events type. Each tweet was part-of-speech (POS) tagged, then named entities and events phrases were extracted and finally the events were categorized into their correct types and visualized as a calendar. TwiCal used a supervised approach by annotating a corpus of 1000 tweets that was used to train a conditional random field classifier to extract events phrases. The TimeBank [2] annotation guidelines were used to annotate their dataset. TimeBank is a corpus that was annotated to indicate events, times, and temporal relations. TimeBank contains the most accurate annotated data of events that existed in the year of 2003. Contextual, dictionary, orthographic and POS features were extracted out of the dataset and used to train the supervised approach. The approach results were compared with a system that did not make use of POS feature, and with another system that was trained on the Timebank corpus and used the same set of features. The F-Score measure for their system was 0.64 compared with a 0.57 to the system that did not use the POS feature and 0.15 to the approach that used the TimeBank corpus.

Becker et al. [6] analyze a stream of tweets of whether a tweet contains an event or not. The approach proposes an online framework that consists of two main tasks: filtering and clustering. By using an incremental clustering algorithm they train a classifier that predicts which cluster should be mapped to a tweet at any point of time. The approach was evaluated using 2,600,000 tweets, which were crawled during February-2010 and compared to the manually annotated 300 clusters test set. Moreover, the approach was compared to a traditional classifier - i.e. Naïve Bayes classifier - where the F-measure for the Naïve Bayes was 0.70 and 0.873 for the streaming approach.

Another work was done by a group of researchers at Yahoo labs, Popescu et al. [9] propose what they called an 'aboutness' system that relies on a huge dictionary of millions of phrases and lexical variants. Using a computational equation, the approach extracts the main events with potential entities that might exists in the tweet. The approach was evaluated on a dataset that consists of 5,040 tweets. The dataset was manually annotated into two main types of tweets, events (2249 tweets) and non-events (2791 tweets), Two main approaches were developed



Fig. 1: Example for an Arabic tweet and its POS tags, the same tweet is tagged with the event arguments.

(a) a regular term frequency inverse document frequency (TF-IDF) based system and (b) the ‘aboutness’ system. The ‘aboutness’ system achieved F-measure value of 0.67 whereas the regular TF-IDF system achieved a 0.66 F-measure score.

Focusing on Arabic data, Alsaedi and Pete [14] train a Naïve Bayes classifier on Arabic tweets to extract events as part of a framework to apply a supervised approach to classify, cluster and summarize events. Worth noting that, this research is the only one available for extracting events from Arabic tweets.

In contrast to previous work on event extraction from Arabic text in general and from Arabic noisy text in particular, our approach focuses on extracting events with their related entities from Arabic tweets and link them to their corresponding entities in the event knowledge base. This enables our approach to provide appealing interactive interfaces of events such as a calendar or a time-line. Moreover, maintaining a knowledge base of events that can be used for event temporal resolution when more than one tweet are referring to the same event using different temporal expressions. Additionally, our approach is extracting events independently of the event type or event domain which makes it more applicable. Adopting an unsupervised technique based on syntactic rules fosters our approach to be more scalable.

### III. PROPOSED APPROACH

For each tweet, proposed approach extracts named entities associated with their event phrases and specific events arguments. Event expressions may consist of the following arguments:

- **Event Agent:** represent the event initiator and the event participants.
- **Event Location:** refers to the city, country, or continent of the event location.
- **Event Target:** where the event is taking place within the event location (e.g. name of an organization, or facility).
- **Event Trigger:** the linguistic expression of the tweet that refers to the event expression.
- **Event Product:** in some cases the event is announcing a product.
- **Event Time:** is the tweet expression indicating the date/time of the event.

Figure 1 depicts an example of an Arabic tweet for a movie screening event with its annotated event arguments. The proposed knowledge-based approach extracts all the event arguments (discussed earlier) if they are mentioned in the text.



three possible syntactic cases can be applied to extract the event trigger, These rules are:

- a **VB +NN:** If the next word is Noun(NN), then the event trigger will be the verb word followed by the noun word, in the above example the event trigger will be (“اعلان/فيلم”/Movie announcement).
- b **VB+VB:** If the next word is a verb (VB/VBN), then the event trigger will be the first verb with the second verb. For instance, “ساره حاولت قتل ماري / sArh HAWlt qtl mAry /Sara Tried to Kill Marry” (حاولت/ HAWlt /Tried) is a verb and (قتل/ qtl/Kill) is also a verb so the event will be (حاولت قتل / HAWlt qtl/Tried to kill).

In case of none of these two rules has applied to the tweet, the event trigger will be only the extracted verb (VB). For example “الى جوبا في كانون الاول- ديسمبر” / frns24: zEym mtmrdrj jnwb AlswdAn yEln bAnh syEwd Ala jwbA fy kAnwn AlAwl-dysmbr /France24: The leader of south Sudan Rebels announced that he will return back to Juba in December. ”, the tool will fail to extract any event trigger using the above rules, so the verb (يعلن / yEln/ announce) will be extracted as the event trigger.

- If the tweet has a Noun (NN or NNP) tag, at this case we have two possible rules:
  - a **NN+NN:** if the next word or next two words are also a Noun such as “الى السبت الى الارباء عطله عيد الاضحى من الارباء / ETlh Eyd AlADHA mn AlArbEa Ala Alsbt /Eid Adha Holydays from Wednesday to Saturday”, then the event trigger will be “عيد الاضحى عطله / ETlh Eyd AlADHA /Eid Adha Holydays” as the POS for “عيد/NN الاضحى/NNP”.
  - b **NN/NNP Only:** At some cases the event trigger could be only a Noun. For example “العطله من الارباء الى السبت /AIEtlh mn AlArbEA Ala Alsbt/The Holydays from Wednesday to Saturday/” here the Noun is (العطله / AIEtlh /Holydays) so the event trigger will be (العطله/AIEtlh /Holydays).
  - c **NN/NNP + VB/VBP:** At some cases the event trigger could be only a verb. For example “حفلي سوف تقام يوم الثلاثاء / Hftly swf tqAm ywm AlvAvA /My party will be held on Tuesday”, this sentence has a verb or a verb-phrase (تقام / will be held) which is proceeded by NN or NNP (حفلي/ My party/ Hftly). In such cases the VB/VBP (تقام / will be held/ tqAm) is used as the event trigger.

In order to extract the event time, the following rules have been used:

- If we have a Cardinal number (CD ) tag for any part of the words of the text there are two possible syntactic rules to extract the event time from the text, These rules are:
  - a **CD + DTNN:** If the next word is a

Determiner (DTNN), then the event time will be the CD+DTNN words. For example “موعد الحفل 5 ايار / mwEd AIHfl 5 AyAr /the concert time is on 5 May “, the time trigger will be (5 ايار AyAr/ 5 May) as the POS tag for “5” is CD and for (ايار / AyAr /May) is DTNN.

- b **CD + DTNN +CD:** If the next word is a (DTNN) and the next-next word is a (CD), then the Time trigger will be the three words together (CD + DTNN + CD). For example (سوف تقام المباراه في 5 ايار 2016 / swf tqAm AlmbArAh fy 5 AyAr 2016 / The match will be held on 5th of May 2016) the time trigger will be (5 ايار 2016 / 5 AyAr 2016/ 5 May 2015) as “5” is tagged as CD, (ايار/ AyAr /may) is tagged as DTNN, and “2016” is tagged as CD.

TABLE I: Syntactic rules for Event Trigger extraction.

Tweet POS Tag	Event Trigger
Verb (VB)/ Verb phrase (VBP)	VB+NN
	VB+VB
Noun (NN) / Noun phrase(NNP)	NN+NN
	NN/NNP Only
	NN/NNP + VB/VBP
Only Verb (VB)	VB
Cardinal number (CD)	CD + DTNN
	CD + DTNN +CD

Tables I summarize the syntactic rules that were used to extract the events triggers. For defining the event type (i.e. instant or interval), the system checks whether the tweet text has more than one event time and/or specific temporal expressions (keywords). Table II gives some examples for both types. For instance “من الساعة 5 الى الساعة 8” / tqAm AIHflh Almwsyqyh mn AlSAEH 5 Ala AlSAEH 8 /The concert will be held from 5 to 8 o'clock”, two times represented in hours are provided in this tweet and are separated by the time keyword (من - الى / from - to) which indicates an event with an interval type.

2) *Named Entity Recognition and Disambiguation:* For all the tweets that have been detected to hold an event, the system tags the missing events’ arguments represented by (a) Event Agent, (b) Event Location and (c) Event product. The event agent can be represented by the persons or organizations participating / affected by the event, whereas the event location holds information about the place of the event. These arguments are represented as entity mentions in the tweet text. Therefore, in this step a named entity recognizer is used to tag these entities.

Extracting the entities mentioned in the event text is not enough, the approach requires linking the discovered entities with their corresponding in the knowledge base. Hence named entity disambiguation step is required. To this end, we employed the tool provided by [22] where a hybrid approach using machine learning and linked data is used to tag and disambiguate entities of the types: person, location, and organization. The disambiguation procedure in this tool is done based on information extracted from Arabic/English Wikipedia and DBpedia graph knowledge base. The tool was tested on a dataset that consist of

TABLE II: Some of the keywords used to indicate event type.

Event Type	Keywords Used
Instant	“تمام الساعة/at specific time/tmAm AlsAEh”, “يوم محدد/specific day/ywm mHdd”, etc.
Interval	“الفترة/period/“Alftra”, “يستمر/continue/ystrmr”, “خلال/through/xlAl”, etc.

over 10k entity mentions from different domains such as; technology, sport and politics. The results shows that the approach was able to correctly annotate 8,494 entity mentions out of 10,068 entity mentions, with an accuracy of 84% on the over whole dataset. Moreover the results based on the entity type were: 76% for the Person entities, 94% for the Location entities, and 78% for the Organization entities [cite Omar paper]. This hybrid approach has been also extended to disambiguate (link) extracted events to their corresponding events in the knowledge base as part of the temporal resolution step.

3) *Temporal resolution*: After extracting events and their entities, an entity linking step is needed. During this phase event arguments are linked to their corresponding entities in the knowledge base. This ensures better integration of extracted information with available ones, and eliminates duplicates from the knowledge base. Moreover, an event can be mentioned in tweets using different ways. For instance the same calendar date can be represented using different event time expressions, for example “next month” and “December, 2015” or “next Monday” and “December 16th” can all refer to the same calendar date based on the date of tweet writing. Entity linking is used to resolve the temporal expressions extracted out of the tweets. For each extracted event, the event arguments are linked to their corresponding in the knowledge base. A new event is populated to the knowledge base if and only if its arguments are not already represented there.

#### IV. EVALUATION AND RESULTS

The results are obtained out of evaluating the proposed approach accuracy on three tasks namely: (T1) event trigger extraction, (T2) event time extraction, and (T3) event type identification. Evaluation results are measured using the approach accuracy where the number of the correctly predicted values for the three tasks are divided by the overall number of tweets (i.e. 1000). Out of the 1000 tweets of events, 122 events were of the type interval and 878 events were of the type instant.

TABLE III: Evaluation results for the three tasks using the accuracy measure.

Task	Accuracy
T1: Event Trigger Extraction	75.9%
T2: Event Time Extraction	87.5%
T3: Event Type Identification	97.7%

As presented in Table III, for T1: event trigger extraction, the approach managed to extract correctly 759 event triggers out of 1000 (Accuracy = 75.9%), whereas for T2: event time extraction, 875 event times were correctly extracted out of 1000 (Accuracy = 87.5%), and finally for T3: event type identification, 977 event types were correctly classified out of 1000 tweets (Accuracy = 97.7%).

TABLE IV: Accuracy for the three tasks based on the event type.

Task	Event Type	
	Instant	Interval
T1: Event Trigger Extraction	75.5%	78.6%
T2: Event Time Extraction	86.7%	92.6%
T3: Event Type Identification	97.9%	95.9%

In order to focus on the results in more detail, extraction results based on event type were evaluated. Table IV presents the approach accuracy results for the three tasks based on each event type. For the instant event type, T1: event trigger extraction accuracy is (663 / 878 = 75.5%), T2: Event time extraction accuracy is (762 / 878 = 86.7%), and T3: event type accuracy is (860 / 878= 97.9%). Whereas for the interval event type, T1: event trigger extraction accuracy is (96 / 122= 78.6%), T2: Event time extraction accuracy is (113/ 122= 92.6%), and T3: event type accuracy is (117 correct / 122= 95.9%).

#### V. DISCUSSION

After analyzing the approach results on the three tasks, the following limitations can be summarized:

- Errors in the POS tagging: for some cases the tags assigned to the tweet tokens were incorrect. This leads into having a violation in some of the syntactic rules and yields to an incorrect extraction especially in the first two tasks, event trigger extraction and event time extraction.
- Conflict between rules: this case happens when two or more syntactic rules are fired. This problem mainly happens in the first two tasks, event trigger extraction and event time extraction. For example “الفجيرة الظفرة على ستاد الفجيرة الرياضي ضمن دوري الخليج العربي” / تنطلق في تمام الساعة 5 مباراة 5 mbArA Alfjyr AlZfr Ela stAd Alfjyr AlryADy Dmn dwry Alxlyj AlErby/It kicks off at 5 pm match Fujairah and the Fujairah Dhafra Sports Stadium within the Arabian Gulf League”, in this case, the system extracts “ستاد الفجيرة / stAd Alfjyr / Fujairah Dhafra Sports Stadium” as the event trigger uses the NN+NN syntactic rule whereas the correct event trigger should follow the rule VB and extracts (تنطلق / tnTlq / kicks off ) as the right event trigger.

The two problems mentioned earlier happen due to the limitations in the used POS tagger. The used POS tagger uses a phrase structure (PS) instead of dependency structure (DS) when annotating sentences. The difference between the two approaches is mainly in the POS tree where the words in the PS are the leaves and syntactic categories such as noun phrase (NP) and verb phrase (VP) represents the internal nodes, whereas in the DS the



tree nodes are represented using the sentence words [31]. The problem is discussed in more details in [32]. Possible solution could be through using a DS-based POS tagger such as The Columbia Arabic Tree Bank (CATiB) [33], [34] or The Prague Arabic Dependency Treebank (PADT) [35], [36].

## VI. CONCLUSION AND FUTURE WORK

In this paper we present a knowledge-based approach for extracting events out of Arabic tweets. The approach uses an unsupervised rule-based approach for Event extraction and a Named entity disambiguation system to map each entity mention to their corresponding entities that are represented in the knowledge base. Results show that the approach has an accuracy of, 75.9% for T1: event trigger extraction, 87.5% for T2: event time extraction, and 97.7% for T3: event type identification.

Regarding enhancing the results of the rule-based approach we plan to use dependency structure based POS tagger - such as PADT and CATiB - instead of the current applied one. DS-based POS taggers builds on both syntactic and semantic features while tagging words, for instance they have semantic tags for time (TMP) and location (LOC). Moreover, DS-based POS taggers are capable to tag the head of the sentence which is most of the time the same event trigger to be extracted.

As for future plans we are currently annotating manually the whole 3K dataset of tweets where part of it will be used to train a supervised approach to extract event triggers and types using sequence labeling techniques such as Conditional Random Fields [35, 5]. Other classifiers can also be used to evaluate the features of POS and NER and results will be compared to the proposed rule-based approach.

## REFERENCES

- [1] J. Allan, R. Papka, and V. Lavrenko, "On-line new event detection and tracking," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1998, pp. 37–45.
- [2] J. Pustejovsky, P. Hanks, R. Sauri, A. See, R. Gaizauskas, A. Setzer, D. Radev, B. Sundheim, D. Day, L. Ferro *et al.*, "The timebank corpus," in *Corpus linguistics*, vol. 2003, 2003, p. 40.
- [3] E. Gabrilovich, S. Dumais, and E. Horvitz, "Newsjunkie: providing personalized newsfeeds via analysis of information novelty," in *Proceedings of the 13th international conference on World Wide Web*. ACM, 2004, pp. 482–490.
- [4] N. Chambers and D. Jurafsky, "Template-based information extraction without the templates," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 976–986.
- [5] C. X. Lin, B. Zhao, Q. Mei, and J. Han, "Pet: a statistical model for popular events tracking in social communities," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 929–938.
- [6] H. Becker, M. Naaman, and L. Gravano, "Beyond trending topics: Real-world event identification on twitter," 2011.
- [7] E. Benson, A. Haghighi, and R. Barzilay, "Event discovery in social media feeds," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 389–398.
- [8] J. Lin, R. Snow, and W. Morgan, "Smoothing techniques for adaptive online language models: topic tracking in tweet streams," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 422–429.
- [9] A.-M. Popescu, M. Pennacchiotti, and D. Paranjpe, "Extracting events and event descriptions from twitter," in *Proceedings of the 20th international conference companion on World wide web*. ACM, 2011, pp. 105–106.
- [10] A. Ritter, O. Etzioni, S. Clark *et al.*, "Open domain event extraction from twitter," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 1104–1112.
- [11] S. Abuleil, "Using nlp techniques for tagging events in arabic text," in *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on*, vol. 2. IEEE, 2007, pp. 440–443.
- [12] I. Saleh, L. Tounsi, and J. van Genabith, "Zaman and raqm: extracting temporal and numerical expressions in arabic," in *Information Retrieval Technology*. Springer, 2011, pp. 562–573.
- [13] H. Aliane, W. Guendouzi, and A. Mokrani, "Annotating events, time and place expressions in arabic texts," in *RANLP*, 2013, pp. 25–31.
- [14] N. Alsaedi and P. Burnap, "Arabic event detection in social media," in *Computational Linguistics and Intelligent Text Processing*. Springer, 2015, pp. 384–401.
- [15] J. M. Pawlowski, M. Bick, R. Peiln, S. Thalmann, R. Maier, D.-W.-I. L. Hetmank, D.-W.-I. P. Kruse, M. Martensen, and H. Pirkkalainen, "Social knowledge environments," *Business & Information Systems Engineering*, vol. 6, no. 2, pp. 81–88, 2014.
- [16] T. Berners-Lee, J. Hendler, O. Lassila *et al.*, "The semantic web," *Scientific american*, vol. 284, no. 5, pp. 28–37, 2001.
- [17] T. Hasegawa, S. Sekine, and R. Grishman, "Discovering relations among named entities from large corpora," in *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2004, p. 415.
- [18] W. Shen, J. Wang, P. Luo, M. Wang, and C. Yao, "Reactor: a framework for semantic relation extraction and tagging over enterprise data," in *Proceedings of the 20th international conference companion on World wide web*. ACM, 2011, pp. 121–122.
- [19] D. Ahn, "The stages of event extraction," in *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*. Association for Computational Linguistics, 2006, pp. 1–8.
- [20] B. Sundheim, "Third message understanding evaluation and conference (muc-3): Phase 1 status report." in *HLT*, 1991.
- [21] M. Althobaiti, U. Kruschwitz, and M. Poesio, "Aranlp: a java-based library for the processing of arabic text," 2014.
- [22] M. Al-Smadi, B. Talafha, O. Qawasmeh, M. N. Alandoli, W. A. Hussien, and C. Guetl, "A hybrid approach for arabic named entity disambiguation," in *Proceedings of the 15th International Conference on Knowledge Technologies and Data-driven Business*. ACM, 2015, p. 27.
- [23] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer *et al.*, "Dbpedia—a large-scale, multilingual knowledge base extracted from wikipedia," *Semantic Web*, vol. 6, no. 2, pp. 167–195, 2015.
- [24] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, *Dbpedia: A nucleus for a web of open data*. Springer, 2007.
- [25] F. Mahdisoltani, J. Biega, and F. Suchanek, "Yago3: A knowledge base from multilingual wikipedias," in *7th Biennial Conference on Innovative Data Systems Research*. CIDR Conference, 2014.
- [26] Y. Raimond and S. Abdallah, "The event ontology," Citeseer, Tech. Rep., 2007.

- [27] J. R. Hobbs and F. Pan, "An ontology of time for the semantic web," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 3, no. 1, pp. 66–85, 2004.
- [28] Y. Raimond, S. A. Abdallah, M. B. Sandler, and F. Giasson, "The music ontology," in *ISMIR*. Citeseer, 2007, pp. 417–422.
- [29] D. Brickley and L. Miller, "Foaf vocabulary specification 0.98," *Namespace document*, vol. 9, 2012.
- [30] "Ace (automatic content extraction)arabic annotation guidelines for entities," 2008.
- [31] F. Xia, O. Rambow, R. Bhatt, M. Palmer, and D. Misra Sharma, "Towards a multi-representational treebank," *LOT Occasional Series*, vol. 12, pp. 159–170, 2008.
- [32] N. Y. Habash, "Introduction to arabic natural language processing," *Synthesis Lectures on Human Language Technologies*, vol. 3, no. 1, pp. 1–187, 2010.
- [33] N. Habash and R. M. Roth, "Catib: The columbia arabic treebank," in *Proceedings of the ACL-IJCNLP 2009 conference short papers*. Association for Computational Linguistics, 2009, pp. 221–224.
- [34] N. Habash, R. Faraj, and R. Roth, "Syntactic annotation in the columbia arabic treebank," in *Proceedings of MEDAR International Conference on Arabic Language Resources and Tools, Cairo, Egypt, 2009*.
- [35] J. Hajic, B. Vidová-Hladká, and P. Pajas, "The prague dependency treebank: Annotation structure and support," in *Proceedings of the IRCS Workshop on Linguistic Databases, 2001*, pp. 105–114.
- [36] J. Hajic, O. Smrz, P. Zemánek, J. Šnidauf, and E. Beška, "Prague arabic dependency treebank: Development in data and tools," in *Proc. of the NEMLAR Intern. Conf. on Arabic Language Resources and Tools, 2004*, pp. 110–117.