Towards Monitoring of Novel Statements in the News

Michael Färber(✉), Achim Rettinger, and Andreas Harth

Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
{michael.faerber,rettinger,harth}@kit.edu

Abstract. In media monitoring users have a clearly defined information need to find so far unknown statements regarding certain entities or relations mentioned in natural-language text. However, commonly used keyword-based search technologies are focused on finding relevant documents and cannot judge the novelty of statements contained in the text. In this work, we propose a new semantic novelty measure that allows to retrieve statements, which are both novel and relevant, from natural-language sentences in news articles. Relevance is defined by a semantic query of the user, while novelty is ensured by checking whether the extracted statements are related, but non-existing in a knowledge base containing the currently known facts. Our evaluation performed on English news texts and on CrunchBase as the knowledge base demonstrates the effectiveness, unique capabilities and future challenges of this novel approach to novelty.

Keywords: Semantic novelty measures · Novelty detection · Statement extraction

1 Motivation

End users – both in a private or professional setting – increasingly face the challenge to screen and analyze large amounts of natural language text in order to find novel statements which they were not aware of previously. Consider for example a Web user who is interested in the latest technical achievements in the smart phone domain. Also, a stock broker might look for acquisitions of certain companies mentioned in the news. When using current technology, all potentially relevant text documents are first roughly selected by keyword search, before being checked manually for statements which are novel to the user.

In this work, we propose an approach to support the task of novel statement detection by automatically extracting so far unknown statements from natural language sentences. There exist numerous techniques for information extraction (IE) on text, i.e. systems which convert sentences into formal representations such as triples or other n-ary relational data. However, the detection of genuinely new facts in text differs from traditional web search or monitoring systems, since typically only relevance is taken into account as a selection

H. Sack et al. (Eds.): ESWC 2016, LNCS 9678, pp. 285–299, 2016.
DOI: 10.1007/978-3-319-34129-3_18

criterion. All existing novelty detection systems are based on syntactical and statistical techniques and are not able to assess written statements w.r.t. novelty on a semantic level. In contrast, our semantic novelty measure allows to satisfy the user's information need to an extent which could not be achieved before: Our novelty detection system (i) determines the semantic novelty by checking against a background Knowledge Base (KB) containing the current knowledge on this domain, (ii) presents only those statements to the user which are relevant to the user's current individual information need expressed by a semantic query and (iii) intuitively shows the user what the novel aspect in a statement is by assigning it to certain novelty classes.

Our proposed novelty search system is domain independent and can be applied in different settings ranging from news monitoring and news summarization to evaluating human generated summaries of documents for completeness. Our main contributions are:

- providing a semantic measure for the novelty of statements based on a background KB,
- proposing a semantic novelty detection system for statements in text documents,
- performing an evaluation of our approach on real-world data to demonstrate its unique capabilities in media monitoring tasks, like forecasting of facts, KB population and impact quantification.

The remainder of this paper is organized as follows: First we present our definition of a semantic novelty measure for facts in Sect. 2, before proposing our semantic novelty detection system in Sect. 3. After discussing our evaluation in Sect. 4 and giving an overview of related work in Sect. 5, we conclude in Sect. 6.

2 Measuring Semantic Novelty of Statements

In this work we assume that each statement[1] we are interested in, i.e. we want to extract from the text, can be represented as a Resource Description Framework (RDF) triple (s, p, o) where $p \in U$ is the binary relation between a subject $s \in U$ and an object $o \in U$, where U is the set of unique identifiers.[2] We further assume that there is a KB represented in RDF which contains all knowledge in the domain of interest of the user. This KB acts as a reference point to assess novelty. There are several tools for personalized knowledge management that produce RDF, like semantic wikis. Also, the constantly growing knowledge graphs like Linked Data sources such as DBpedia as well as company-internal knowledge graphs (Google's Knowledge Graph, Microsoft's Satori, etc.) can be used as an initial collection of knowledge.

In our scenario, the user wants to retrieve triples which are both relevant and novel (see Fig. 1) and which can be added automatically to his KB.

[1] We use the words "statements", "facts", and "triples" interchangeably in this paper.
[2] We do not consider triples where the object is a literal.

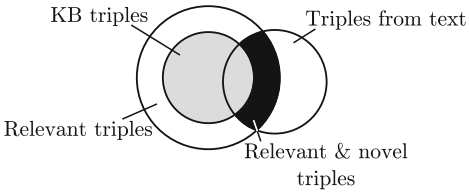


Fig. 1. Triples relevant to the user, KB triples, and triples extracted from text visualized as different sets in a Venn diagram. The aim of our novel triple extraction system is to retrieve triples which are novel (not yet in the KB), but relevant (black area).

Our three-step filtering process facilitates this: Firstly, all extracted triples must be relevant according to the given user query. Secondly, all triples have to be novel in the sense that they do not occur in the KB yet. Thirdly, the novel, but relevant triples need to complement the existing knowledge seamlessly. This is ensured by requiring the novel statements to partially overlap with the KB:

1. Triples extracted from text where an identical triple exists in the KB are not novel.
2. Triples which consist of two parts – (s, p), (p, o), or (s, o) – existing in the KB and one part (entity or relation) which is not yet in the KB, i.e. unknown, are regarded as novel and contextually fitting, since the extracted facts are related to known elements, but also contain new knowledge.
3. Triples which consist of two or three unknown parts are novel, but are assumed unrelated w.r.t. the KB, since the gap between these extracted triples and triples in the KB is too wide resulting in mostly irrelevant statements.

In Table 1 we formally define the different cases of novelty.

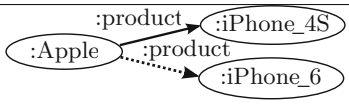
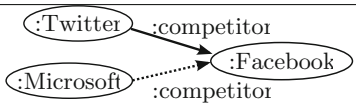
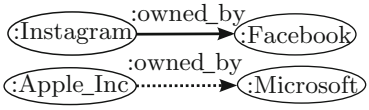
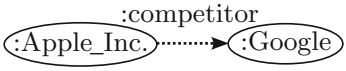
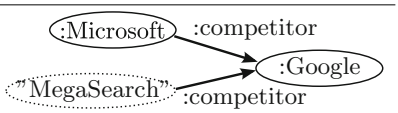
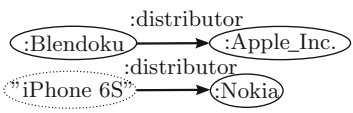
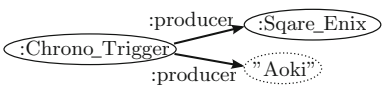
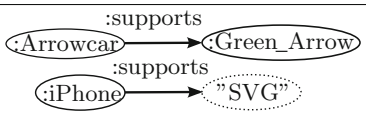
By using this semantic definition of novel facts, we do not limit ourselves to a notion of novelty relying on the creation date of each document to assess whether the information is novel or not [1,2]. Instead, we express the fact as a binary relationship. A change over time is intrinsically expressed via a different structure and/or semantics of the novel triple.

3 The Novel Statement Extraction System

Figure 2 presents an overview of our novel triple extraction system. In principle, we have implemented our system as a three step process with the steps *Textual Triple Extraction*, *KB Linking*, and *Novelty Detection*. In the following, we give a description of each of these steps.

Textual Triple Extraction. In this step each sentence of each input document is transformed into propositions, i.e., statements consisting of a subject, a relation, and none, one, or more arguments (e.g., grammatical direct object). For each proposition found in a sentence and apparently compatible with our RDF knowledge representation, we retrieve so called textual triples (see Fig. 3).

Table 1. The different classes of novel triples with textual descriptions, formal representations, and examples. Dotted lines indicate the novel items in the triple. s_t, p_t, and o_t represent the textual mentions of a subject $s \in U$, predicate $p \in U$, and object $o \in U$, respectively, in a sentence. U is the set of unique identifiers. f is a function which maps a textual resource (s_t or o_t) to its corresponding RDF resource: $f(s_t) = s$ and $f(o_t) = o$. The function g maps a textual predicate p_t to an RDF predicate: $g(p_t) = p$. In cases of unsuccessful matches, we write $\emptyset$.

A	The triple was found and is therefore not novel. $(s, p, o) \in KB$	
$B_{1,i}$	A new additional outgoing relation of s is found. $(s, p, o) \notin KB \wedge \exists x \in U : (s, p, x) \in KB$	
$B_{1,ii}$	A new additional incoming relation of o is found. $(s, p, o) \notin KB \wedge \exists x \in U : (x, p, o) \in KB$	
B_2	A new relation between existing s and o is found. The relation type detected already exists in the KB. $(s, p, o) \notin KB \wedge p \neq \emptyset \wedge \neg\exists x \in U : ((s, p, x) \in KB \vee (x, p, o) \in KB)$	
C	A completely novel relation is found, where only s_t and o_t are matched. $(s, p, o) \notin KB \wedge s \neq \emptyset \wedge o \neq \emptyset \wedge p = \emptyset$	
D_1	Here, s_t could not be matched, but the matched object o has already another incoming relation of the same type. $(s, p, o) \notin KB \wedge \exists x \in U : (x, p, o) \in KB \wedge s = \emptyset$	
D_2	s_t could not be matched to a KB resource, but the relation p_t and object o_t are contained in the KB, although not within one triple. $(s, p, o) \notin KB \wedge p \neq \emptyset \wedge \neg\exists x \in U : (x, p, o) \in KB \wedge s = \emptyset$	
E_1	o_t could not be matched, but the resource s has another outgoing relation of the same type as the matched relation p. $(s, p, o) \notin KB \wedge \exists x : (s, p, x) \in KB \wedge o = \emptyset$	
E_2	o_t could not be matched to any resource o, but the relation is known to exist already in the KB. $(s, p, o) \notin KB \wedge p \neq \emptyset \wedge \neg\exists x \in U : (s, p, x) \in KB \wedge o = \emptyset$	

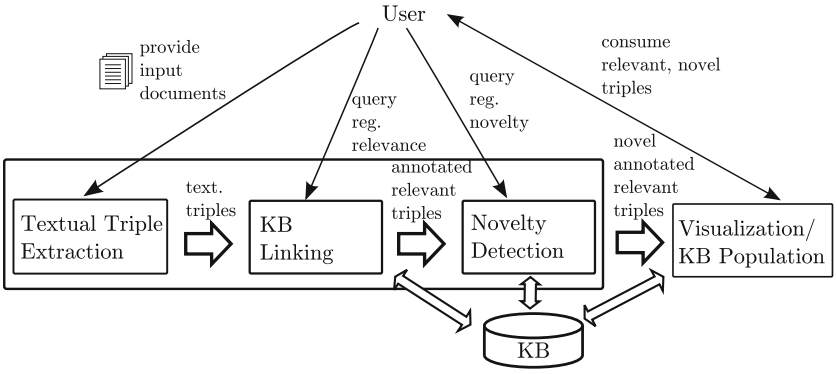


Fig. 2. Overview of our novel triple extraction system with the three steps of (i) *Textual Triple Extraction*, (ii) *KB Linking*, and (iii) *Novelty Detection*.

Textual triples are of the form (s_t, p_t, o_t), where s_t and o_t are natural-language mentions of entities and p_t is a relation found in the sentence. For the example sentence "The sixth generation iPhone was developed by Apple and features a touchscreen display.", we retrieve the textual triples (i) ("Apple", "develop","sixth generation iPhone") and (ii) ("sixth generation iPhone", "feature","touchscreen display").

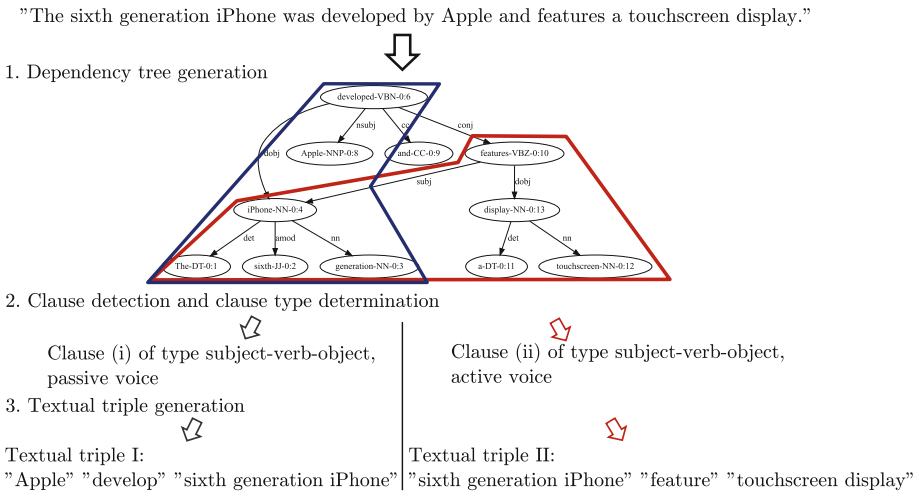


Fig. 3. Different steps of our Textual Triple Extraction module.

Technically, the Textual Triple Extraction step is based on the tool ClausIE [3]. However, we need to modify ClausIE in terms of linguistic processing to our requirements:

- Adjectives cannot be encoded into RDF triples easily and even prevent that the entity mention or predicate can be mapped to the correct KB resource or relation, respectively. We therefore remove all adjectives from the dependency tree.
- Temporal aspects expressed in the input sentence (point in time or period of time) such as "yesterday" are difficult to attach to a triple, but can be represented more adequately as separate information units. Hence, we extract temporal phrases with the help of the inherent Stanford PoS tagger and store them separately.
- In sentences which contain linguistic complements, as they often occur in case of indirect speech ("X said that Y has Z"), we focus on the extraction of the proposed fact itself (which is the complement), not the subject of the sentence (e.g. the person X who said something).
- If a clause in a sentence is written in passive voice, we transform it into active voice.

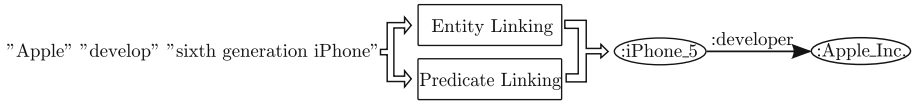


Fig. 4. KB Linking illustrated by an example. Here it is assumed that all parts of the textual triple can be mapped.

KB Linking. In this second step we map the textual phrases s_t, p_t, and o_t to the corresponding KB entities and, respectively, KB properties as far as possible (see Fig. 4). For entity linking we exploit *xLiD lexica* [4] by which we can link textual mentions of an entity to its corresponding resource in the KB DBpedia.[3] Due to the ambiguity problem (e.g., the mention "Apple" can represent the DBpedia entity `:Apple_Inc.`,[4] the fruit `:Apple` and several other entities), we integrate a disambiguation step based on [5] in order to find the most likely entity for each mention. For a sufficient disambiguation, we take all mentions in the current sentence into account.[5]

[3] See http://dbpedia.org, requested on Mar 7, 2016. DBpedia is widely used for entity linking in general domain settings. However, also other KBs can be used as far as a suitable entity linking component is available.

[4] We avoid the DBpedia namespaces for better readability.

[5] For our example sentence (see Fig. 3), this would be "sixth generation iPhone", "Apple", and "touchscreen display".

Furthermore, relations expressed in the text need to be linked to the corresponding KB properties. For instance, the textual property "bought" needs to be linked to the KB property `:acquired`. However, linking the textual predicate p_t to the corresponding KB property p is a hard task [6], since KB properties are typically not equipped with the information how they occur in natural-language texts. Trying to match the label of the KB property (e.g., "acquired") with the textual predicate p_t (e.g., "bought") is generally inefficient. One reason for that is that the KB property can be expressed by a variety of expressions. In our work, we therefore set up a two-step process for predicate linking, which is similar to the approach of [7]: In a *training phase* which is performed before the actual novel triple extraction phase, our system performs the Textual Triple Extraction and the KB Linking. If both the textual subject s_t and object o_t of an extracted textual triple could be mapped to KB entities and if the classes of these KB entities match with the pre-defined domain and range of the target KB property, the corresponding p_t might express the target property p. In a semi-automatic fashion a user then confirms or rejects the found mappings $p_t \rightarrow p$ (e.g., "buy" $\rightarrow$ `:acquired`). In case the mapping is confirmed, it is added to the mappings $p_t \rightarrow p$. At the end we have for all considered KB properties mappings $p_t \rightarrow p$ learned from text. In the application phase, i.e. the actual novel triple extraction phase, the Textual Triple Extraction, the KB Linking, and the Novelty Detection step is run. Here, all novel triples except from novelty class C (see Table 1) are extracted, since novelty class C was used in the training to find mappings from textual predicates to KB properties.

At the end of the KB Linking step, we have textual triples which are mapped to KB triples either partly or completely. Given a semantic user query regarding the relevance of the extracted triples (consisting of basic graph patterns and implemented as SPARQL query; a query expressed in natural language might be:"Retrieve all acquisitions of companies in the smartphone domain."), all triples are filtered out which are irrelevant.

Novelty Detection. We determine which of the remaining triples are novel w.r.t. our KB and classify these triples into the different novelty classes defined in Sect. 2. Our system is designed for allowing the user to choose which novelty classes should be considered by the system. The user query is, hence, extended by the information need regarding the novelty classes (e.g., only triples of the novelty classes $B_{1,i}$, $B_{1,ii}$, and B_2 should be retrieved).

4 Evaluation

In the following, we first present the data sets used for evaluating our approach.[6] We then describe the evaluation settings and finally show the results of our evaluation.

[6] The data sets and evaluation results are available at http://people.aifb.kit.edu/mfa/novel-triple-extraction/.

4.1 Data Used

KB: As KB we used CrunchBase[7] in combination with DBpedia.[8] CrunchBase consists of structured information about organizations (including companies), people, products, investments, and several other items, and is edited by a Web community. We built an RDF KB using the CrunchBase API and integrated `owl:sameAs`-relations to the corresponding entities in DBpedia. By bridging to DBpedia, we can use the existing entity linking module [4,5] for mapping the textual subjects and objects. This approach is more robust than using simple string matching based on the CrunchBase entity labels.

In our evaluation, we focus on relations between organizations and on relations between persons and organizations (see Fig. 5). Our CrunchBase RDF KB contains 16,706 entities of type organization and 26,468 entities of type person – in both cases with corresponding `owl:sameAs`-links to DBpedia. There are 16,509, 60,936, 151,722, and 83,470 distinct facts regarding the KB properties `cb:acquired`, `cb:competesWith`, `cb:founded`, and `cb:isBoardMemberOrAdvisor`, respectively.

Documents: We used English news articles from the IJS newsfeed[9] [8] as input text documents for our novel triple extraction system. For learning the textual predicate mappings in the training phase (cf. Sect. 3), we used all English news from May 1 until May 15, 2015 (607,289 articles) and ignored those paragraphs which contained no known label of a person or organization in our KB (leading to 136,907 paragraphs). For the actual triple extraction phase, we chose the time range of May 16 until August 31, 2015, (3,642,771 articles) and applied the same filter, resulting in 797,224 paragraphs of text.

4.2 Evaluation Setting

Our evaluation addresses the following claims:

1. **Fact Forecast:** We claim that we can detect facts, such as acquisitions, which are sometimes leaked, rumored or discussed publicly before they are officially announced. This is of great interest in media monitoring.
2. **Improved KB Population:** Given true new facts, our system provides a comfortable way to insert these facts to the KB: (i) Our method can detect and extract novel and known facts mentioned in the news faster than if they were added to the KB manually. (ii) Our method already provides new facts in a semantically-structured format, ready for inserting it to a KB. (iii) Our method can provide links to the news articles (together with other meta-data) in which relevant novel facts are mentioned. This provenance information can be added to the KB and provides evidence for the facts (fact verification).

[7] See http://crunchbase.com, requested on Mar 7, 2016.
[8] See http://dbpedia.org, requested on Mar 7, 2016.
[9] See http://newsfeed.ijs.si, requested on Mar 7, 2016.

3. **Impact Quantification:** For all facts stored in the KB, our system can show when and how often these facts have been mentioned in the news. This feature can be used for tracking facts, e.g., in the context of beat reporting.

Hence, our novel triple extraction system was evaluated in two parts:

1. We evaluated whether our system can achieve the above mentioned goals 1, 2, and 3 regarding acquisition facts. The query can be formulated in natural language as: "Extract all novel triples (considering all novelty classes) with the relation `cb:acquired`."
2. In a second evaluation, we expanded the query to retrieve all novel facts with the KB properties `cb:acquired`, `cb:competesWith`, `cb:founded`, and `cb:isBoardMemberOrAdvisor` (see Fig. 5) and evaluated how many facts were correctly extracted from the news, thereby considering goal 2.[10]

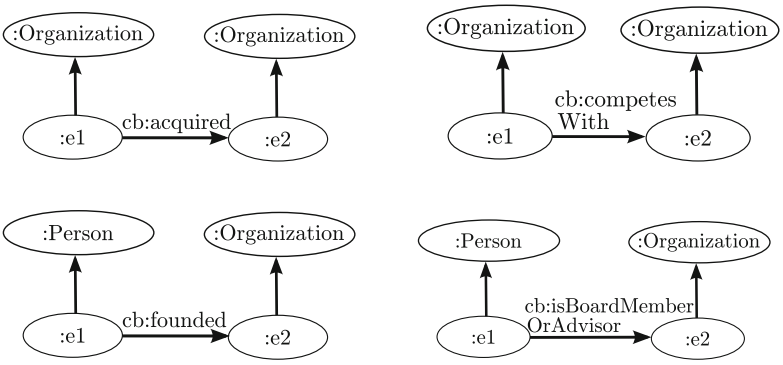


Fig. 5. KB properties with their domains and ranges as used in the second evaluation. `owl:sameAs` links to DBpedia entities and their `rdf:type` relations are not shown for convenience.

4.3 Evaluation Results

Evaluation Part 1

Regarding Fact Forecast: Given the news of the specified time range (May 16 until August 31, 2015; 797,224 paragraphs), our novelty detection tool was able to extract 32 distinct acquisition facts (89 in total) which were also in CrunchBase given the state of October 1, 2015. Out of these 32, two acquisitions (i) were announced within the specified time range according to CrunchBase and

[10] Goal 1 and 3 are not considered here since facts with the chosen KB properties do not occur often.

(ii) were detected by our tool back then before they were inserted into CrunchBase (see Fig. 6). This shows that our system is able to detect facts before they might actually become true. Our manual evaluation on all 1,333 retrieved `cb:acquired` facts (see Fig. 2) revealed that the extracted hypothetically formulated facts about acquisitions can be found across the novelty classes (3 6, 3, 2, 3, 0, 8, and 3 occurrences for the novelty classes A, $B_{1,i}$, $B_{1,ii}$, B_2, D_1, D_2, E_1, and E_2, respectively).

Regarding Improved KB Population: Out of the 32 distinct extracted acquisitions, 4 acquisitions (i) had been announced – according to CrunchBase – and inserted to CrunchBase in the specified time frame of the news, i.e., those triples are really novel (Class B) considering the CrunchBase state from May 15, 2015 and (ii) were not written as hypotheses. In total, 69 acquisitions were announced according to CrunchBase within the specified time range. However, 21 of the 63 not-detected acquisitions were not mentioned at all in the selected news, 34 were mentioned at most five times.[11] The approximative *recall* is therefore at least 6/42=0.143.

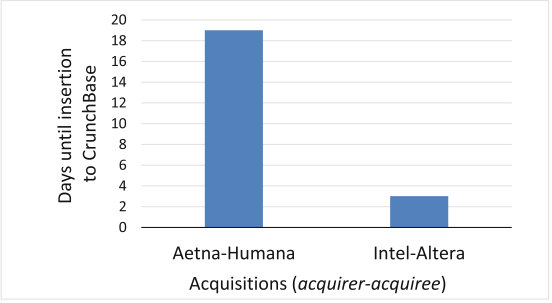


Fig. 6. Days between the retrieval dates of the news where the facts were extracted from and the dates of manual insertion into CrunchBase. Shown are only acquisitions with positive value.

Regarding Impact Quantification: The most repeated acquisitions were `:Facebook :acquired :Oculus VR` (18 times), `:Verizon_Communications :acquired :AOL` (7 times), and `:Apple_Inc. :acquired :Beats_Electronics` (6 times). Figure 7 shows how such repeatedly mentioned facts can be visualized w.r.t. the fact `:Facebook :acquired :Oculus_VR` extracted from the news of March 25–28, 2014.

[11] This analysis was performed by evaluating all sentences containing two labels of the entities of acquisitions which were missed and containing the phrase "acquire"/"acquisition"/"buy"/"purchase" etc.

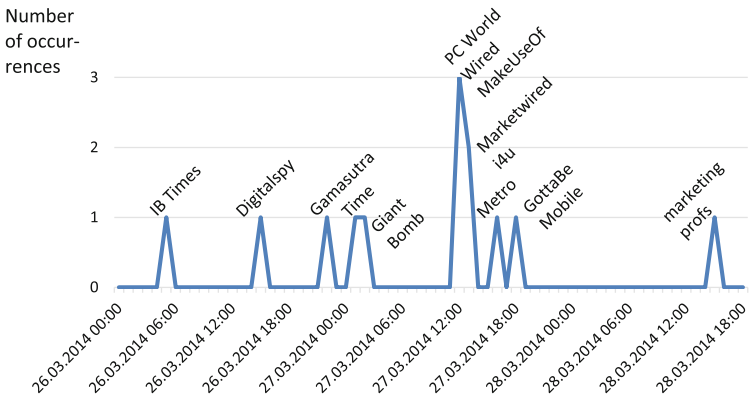


Fig. 7. Publishing dates of the news of March 25–28, 2014, where the fact `:Facebook :acquired :Oculus_VR` was extracted, together with the source such as "IB Times".

Table 2. Number of correct novel triples (as far as evaluated) and number of extracted novel triples per novelty class.

KB property	A	$B_{1,i}$	$B_{1,ii}$	B_2	D_1	D_2	E_1	E_2
`:acquired`	89/89	33/36	4/4	13/13	24/86	14/71	63/636	48/398
`:competesWith`	7/8	35/35	11/13	19/20	72	57	171	240
`:founded`	33/33	0/0	1/2	3/3	63	73	145	311
`:isBoardMemberOrAdvisor`	1/2	7/9	18/18	7/8	58	70	70	450

Evaluation Part 2 – Regarding Improved KB Population: Using the CrunchBase version of October 1, 2015, and the news of the specified time range and not considering the announcement dates, our system was able to retrieve known and novel facts as presented in Table 2. Regarding the 89 extracted facts about acquisitions which were classified as known (Class A), we found out that four of them would have been detected by our tool before they were added to CrunchBase.

The high *precision* of the novel facts of the novelty classes $B_{1,i}$, $B_{1,ii}$, and B_2 (on average $281/293 = 0.959$) shows that especially in cases where the entities are already known (as it is usually the case), our system is able to retrieve high-quality novel facts. The numbers also indicate that CrunchBase misses a significant number of facts.

The manual assessment on the correctness of the extracted novel triples with the relation `:acquired` and with the novelty classes D_1, D_2, E_1 and E_2 revealed a *precision* of $149/1192 = 0.125$. Note, however, that regarding all incorrect triples, about every second triple is almost completely correct (520 occurrences, i.e., 49.8 %). In those cases, all mappings of the annotated triple are correct and the non-mapped part does not contain any ballast. In those cases, the non-mapped part of the triple contains some additional information mentioned in

the text (mostly nominal phrases, e.g., "Chicago-based DTZ") which prevents a mapping. A proposition often resulted in an incorrect triple of Class D or E and simultaneously in a correct triple of Class B. Hence, triples of class D and E may be ignored. Further common failures are that coreferences cannot be resolved (161 cases, i.e., 15.4 %). In 324 cases (31.0 %), although the extracted statement cannot be represented in the triple format or the non-mapped entity was too abstract for being relevant for the KG, the statement was output by the system and, hence, judged as incorrect by the assessor. Determining those cases automatically is non-trivial and left for future work.

5 Related Work

There are several areas of work which are worthwhile to mention as related work (see Table 3 for an overview). First of all, there are approaches to information extraction where the relations and/or entities have a grounding in a knowledge base. They are either based on shallow parsing or deep natural language processing. FRED [9] is one approach of deep NLP where text is transformed into a complex semantic model via Discourse Representation Theory (DRT). In this way, complex statements can be expressed, not only simple binary relations, which we focus on and which makes a comparison with a KB easier. The system of Carvalho [10] is similar to FRED in terms of constructing a structured data graph (SDG). LODifier [11] embraces several existing tools for deep semantic analysis for transforming text into RDF. However, in contrast to our system, the focus of the LODifier is high recall instead of high precision and the tool is designed for scenarios with no a-priori schema information.

Table 3. Summary of related work.

	Extraction of and queries on statements (graph/tupel)	Grounding of extracted entities in a KB	Grounding of extracted relations in a KB	Novelty detection task implemented	Implemented and evaluated on a Web scale
Presutti et al. [9]	✓	✓	✓		✓
Carvalho et al. [10]	✓	✓			
Augenstein et al. [11]	✓	✓	✓		
Fader et al. [12]	✓				✓
Del Corro et al. [3]	✓				✓
Mausam et al. [13]	✓				✓
Zhang et al. [14]				✓	✓
Gabrilovich et al. [1]				✓	✓
Karkali et al. [2]				✓	✓
Li et al. [15,16]				✓	✓
Systems for TREC Novelty Track 2002-2004 [17]				✓	✓
Systems for TREC KBA (2013–2014)	✓			✓	✓
Clarke et al. [18]				✓	✓

Secondly, there are approaches [3,13] based on the idea of Open Information Extraction (OIE). OIE has become prominent as a method for extracting relations in web documents on a huge scale. The aim of OIE is to build a database for textually expressed relations plus associated textually grounded instances without any schema information. RDF – as we need it in our case as final output of our text processing pipeline – is not supported by OIE tools. Converting OIE triples to RDF is non-trivial, however explored by Dutta et al. [19]. As we saw in Sect. 3, we use the tool ClausIE [3] as part of our processing pipeline, but we need to modify it to our requirements.

Our semantic novelty detection approach is, to the best of our knowledge, the first system which introduces a semantic novelty measure by relying on an RDF KB. By formalizing novelty on a triple level, we go beyond pure statistical approaches (often in combination with named entity recognition) [1,2,14–16] for novelty detection.

For the TREC novelty track [17] of the years 2002–2004, systems had to solve different tasks regarding novelty retrieval. The most similar task to our scenario is stated as follows: Given all documents of a broad topic, identify all relevant and novel sentences. Events and opinions form the two types of topics which were provided. In 2004, the number of topics was limited to 50. Each of the 50 topics was defined by a short description and a task narrative.[12] Contrary to that, we generate a formal representation of new statements in terms of an RDF graph and compare novelty on a triple basis. Instead of broad topics, we focus on single relations between entities.

For the subtask *Vital Filtering* of the TREC Knowledge Base Acceleration (KBA) call[13], systems judge the utility of documents mentioning an entity. However, the used tags such as *vital* are not appendant to specific properties of entities. The subtask *Streaming Slot Filling* is about gathering attribute values of specific entities from the text. The set of possible slots and entities is fixed. However, (i) the ground truth for that task in TREC KBA 2014 does not provide information about where in the corpus the slot values were found; (ii) there is no grounding of the slot values, only textual phrases from the text are provided. Regarding the data sets of 2012/2013 and 2015, the TREC Dynamic Domain Track, we face similar differences to our approach. Clarke et al. [18] present an evaluation framework which rewards novelty. Regarding novelty, ranked lists are considered where the relevance of each element is dependent on the proceeding ones.

6 Conclusion and Future Work

Targeted search for novel, formal and grounded facts in unstructured text is an open issue, since existing novelty detection systems primarily regard novelty as

[12] For instance, for the topic "Diana Car Accident", the task was to find novel information about where the accident happened, who was killed, the extent of injuries, how it happened, and who else was involved.

[13] See http://trec-kba.org, requested on Mar 7, 2016.

a statistical filtering step of sentences or documents. In this paper we presented a conceptually new approach that can satisfy the user's information need in a fine grained manner by extracting novel statements in the form of RDF triples. Novelty is hereby measured w.r.t. a background KB and semantic novelty classes.

Our experiments demonstrated that our prototypical system can facilitate (i) *fact forecast*, i.e., detecting hypothetically formulated statements before they are officially announced; (ii) *improved KB population*, i.e., retrieving both novel and relevant facts in a semantically-structured format, with references to the news, potentially even before the fact is inserted manually to the KB by the community; (iii) *impact quantification*, i.e., monitoring the frequency of certain statements over time and thus the impact of this statement.

While this paper constitutes a step towards a more precise and meaningful monitoring of novelty in news, the biggest challenge towards establishing it in a professional setting is to improve recall. A promising next step to achieve this is to improve the Textual Triple Extraction step. This includes a more elaborated textual subject/object extraction (eliminating noise). A further increase of recall could be archived by implementing coreference resolution, so that more subjects and objects can be linked to the corresponding KB entities. Last but not least, recall might be significantly improved by considering relations which are expressed by other means than verbs such as nominalized verbs (e.g., "the acquisition of X by Y"). We believe that the qualitative improvements of our approach justify future research efforts to close the quantitative performance gap to traditional novelty approaches.

Acknowledgement. This work was carried out with the support of the German Federal Ministry of Education and Research (BMBF) within the Software Campus project *SUITE* (Grant 01IS12051).

References

1. Gabrilovich, E., Dumais, S., Horvitz, E.: Newsjunkie: providing personalized newsfeeds via analysis of information novelty. In: Proceedings of the 13th International Conference on World Wide Web, WWW 2004, pp. 482–490. ACM, New York (2004)
2. Karkali, M., Rousseau, F., Ntoulas, A., Vazirgiannis, M.: Efficient online novelty detection in news streams. In: Lin, X., Manolopoulos, Y., Srivastava, D., Huang, G. (eds.) WISE 2013, Part I. LNCS, vol. 8180, pp. 57–71. Springer, Heidelberg (2013)
3. Del Corro, L., Gemulla, R.: ClausIE: clause-based open information extraction. In: Proceedings of the 22nd International Conference on World Wide Web, WWW 2013, Republic and Canton of Geneva, Switzerland, pp. 355–366. ACM (2013)
4. Zhang, L., Färber, M., Rettinger, A.: xLiD-Lexica: cross-lingual linked data lexica. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014), pp. 2101–2105. European Language Resources Association (2014)
5. Zhang, L., Rettinger, A.: X-LiSA: cross-lingual semantic annotation. PVLDB **7**(13), 1693–1696 (2014)

6. Welty, C., Fan, J., Gondek, D., Schlaikjer, A.: Large scale relation detection. In: Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading. FAM-LbR 2010, Stroudsburg, PA, USA, pp. 24–33. Association for Computational Linguistics (2010)
7. Gerber, D., Ngonga Ngomo, A.C.: Bootstrapping the linked data web. In: 1st Workshop on Web Scale Knowledge Extraction @ ISWC 2011 (2011)
8. Trampuš, M., Novak, B.: Internals of an aggregated web news feed. In: Proceedings of the Fifteenth International Information Science Conference IS SiKDD 2012, pp. 431–434 (2012)
9. Presutti, V., Draicchio, F., Gangemi, A.: Knowledge extraction based on discourse representation theory and linguistic frames. In: ten Teije, A., Völker, J., Handschuh, S., Stuckenschmidt, H., d'Acquin, M., Nikolov, A., Aussenac-Gilles, N., Hernandez, N. (eds.) EKAW 2012. LNCS, vol. 7603, pp. 114–129. Springer, Heidelberg (2012)
10. Carvalho, D.S., Freitas, A., da Silva, J.C.P.: Graphia: extracting contextual relation graphs from text. In: Cimiano, P., Fernández, M., Lopez, V., Schlobach, S., Völker, J. (eds.) ESWC 2013. LNCS, vol. 7955, pp. 236–241. Springer, Heidelberg (2013)
11. Augenstein, I., Padó, S., Rudolph, S.: LODifier: generating linked data from unstructured text. In: Simperl, E., Cimiano, P., Polleres, A., Corcho, O., Presutti, V. (eds.) ESWC 2012. LNCS, vol. 7295, pp. 210–224. Springer, Heidelberg (2012)
12. Fader, A., Soderland, S., Etzioni, O.: Identifying relations for open information extraction. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. EMNLP 2011, Stroudsburg, PA, USA, pp. 1535–1545. Association for Computational Linguistics (2011)
13. Mausam, S., M., Bart, R., Soderland, S., Etzioni, O.: Open language learning for information extraction. In: Proceedings of the 2012 Joint Conference on Empirical Methods in NLP and Computational Natural Language Learning. EMNLP-CoNLL 2012, Stroudsburg, PA, USA, pp. 523–534. ACL (2012)
14. Zhang, Y., Callan, J., Minka, T.: Novelty and redundancy detection in adaptive filtering. In: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR 2002, pp. 81–88. ACM, New York (2002)
15. Li, X., Croft, W.B.: An information-pattern-based approach to novelty detection. Inf. Process. Manag. **44**(3), 1159–1188 (2008)
16. Li, X., Croft, W.B.: Novelty detection based on sentence level patterns. In: Proceedings of the 14th ACM International Conference on Information and Knowledge Management. CIKM 2005, pp. 744–751. ACM, New York (2005)
17. Soboroff, I., Harman, D.: Novelty detection: the trec experience. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Vancouver, British Columbia, Canada, pp. 105–112. Association for Computational Linguistics (2005)
18. Clarke, C.L., Kolla, M., Cormack, G.V., Vechtomova, O., Ashkan, A., Büttcher, S., MacKinnon, I.: Novelty and diversity in information retrieval evaluation. In: Proceedings of the 31st Annual International ACM SIGIR Conference on R&D in Information Retrieval. SIGIR 2008, pp. 659–666. ACM, New York (2008)
19. Dutta, A., Meilicke, C., Stuckenschmidt, H.: Semantifying triples from open information extraction systems. In: STAIRS 2014 : Proceedings of the 7th European Starting AI Researcher Symposium, IOS Press, pp. 111–120, Clifton, VA (2014)