

Article

Event Detection in Wikipedia Edit History Improved by Documents Web Based Automatic Assessment

Marco Fisichella ^{1,*}  and Andrea Ceroni ²¹ L3S Research Center, Leibniz University of Hannover, 30167 Hannover, Germany² Joblift, 10437 Berlin, Germany; ceroni@L3S.de

* Correspondence: mfishichella@L3S.de

Abstract: A majority of current work in events extraction assumes the static nature of relationships in constant expertise knowledge bases. However, in collaborative environments, such as Wikipedia, information and systems are extraordinarily dynamic over time. In this work, we introduce a new approach for extracting complex structures of events from Wikipedia. We advocate a new model to represent events by engaging more than one entities that are generalizable to an arbitrary language. The evolution of an event is captured successfully primarily based on analyzing the user edits records in Wikipedia. Our work presents a basis for a singular class of evolution-aware entity-primarily based enrichment algorithms and will extensively increase the quality of entity accessibility and temporal retrieval for Wikipedia. We formalize this problem case and conduct comprehensive experiments on a real dataset of 1.8 million Wikipedia articles in order to show the effectiveness of our proposed answer. Furthermore, we suggest a new event validation automatic method relying on a supervised model to predict the presence of events in a non-annotated corpus. As the extra document source for event validation, we chose the Web due to its ease of accessibility and wide event coverage. Our outcomes display that we are capable of acquiring 70% precision evaluated on a manually annotated corpus. Ultimately, we conduct a comparison of our strategy versus the *Current Event Portal* of Wikipedia and discover that our proposed WikipEvent along with the usage of Co-References technique may be utilized to provide new and more data on events.

Keywords: Wikipedia; user edits; event detection; event validation; temporal retrieval; clustering



Citation: Fisichella, M.; Ceroni, A. Event Detection in Wikipedia Edit History Improved by Documents Web Based Automatic Assessment. *Big Data Cogn. Comput.* **2021**, *5*, 34. <https://doi.org/10.3390/bdcc5030034>

Academic Editors: Michael Alexander Riegler and Pål Halvorsen

Received: 1 July 2021
Accepted: 30 July 2021
Published: 4 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Wikipedia can be considered the biggest online multiple languages encyclopedia. Its widespread extent and good quality of facts elevate Wikipedia as a famous source of information in numerous study topics. Studies that utilize Wikipedia have attracted a lot of research interest over the last years, together with know-how discovery and management, NLP, social-network behavior examine, and so on. A lot of present research consider Wikipedia as a non-dynamic collection, i.e., the saved records are stable or are hardly ever modified. This is one of the most relevant features of Wikipedia that renders it so successful [1]. However, in reality, Wikipedia grows very rapidly, with new pages added and modified regularly by a massive global network of engaged participants. This provides incentive for an effective method to research and retrieve information, with attention toward chronological dynamics.

We cope with the hassle of retrieving information from Wikipedia complex event structures, together the entities that are related at a given chronological period.

The edit history in Wikipedia represents the full evolution of a pages' content. Our technique has the strength of being able to be applied to any language due to the fact that it is currently independent from the range of entities to be acknowledged a priori. In fact, our approach can retrieve occasions with easy structure (e.g., the publishing of a *new book*) to those with less simple structures (e.g., a *demonstration*). Our technique clearly uses the dynamics of facts (events) in Wikipedia; as a result, it can come across numerous

events concerning many articles, because the articles' content changes over the years. In an evaluation to preceding studies detecting events from Wikipedia [2], our approach requires no training information.

Several challenges arise while detecting dynamic relationships and connected events. First, these relationships do not rely on any pre-existing schema. Second, the underlying facts often expose a non-fixed duration that is known in advance: One event can remain for a short time and, at the same time as others, may endure over numerous weeks or months. Third, the entities show a vivid flexibility in participating on events: A few facts can involve two entities and, at the same time as others, are amongst multiple entities [3]. Fourth, the internet network pushes itself to file an actual-existence fact as it occurs. Some data generated in a specific chronological window will no longer be reported in future versions of the pages containing the entities participating in that fact. For that reason, users need the opportunity to obtain access to historic information.

Manually assessing if a fact takes place in a collection is an exhausting assignment, which calls for a manual inspection of textual content. Furthermore, it will become unfeasible in domains wherein facts are continuously and automatically retrieved on a big-scale. We suggest automatically conducting event validation by the method of gaining knowledge of events in a given non-annotated corpus. Considering a set of facts, our technique can (i) reduce the variety of false facts within the given corpus and (ii) discover documents to confirm the presence of facts and refine existing knowledge bases (e.g., [4]). We mainly address the previous use cases by introducing event validation as a step conducted after the event retrieval to refine the precision inside the detected set of facts without negatively impacting on recall. Due to the fact that the presence of an event depends on the documents in a corpus, we distinguish two types of validation: (i) *document-level validation*, which proves the presence of a fact in a given report; (ii) *corpus-level validation*, which proves the presence in the whole corpus. In order to check facts, we expand a supervised learning model by means of extracting specific features from facts and documents. Our version for event check does not require any assumptions on the character of facts and documents, and thus it can be carried out on a huge variety of facts and related corpora. Validity annotations for training our model are received by the method of exploiting the crowd-sourcing paradigm [5].

In this study, we present the following contributions:

1. A generic approach agnostic to linguistic constraints adapted for multi lingual Social Media.
2. Adaptation, formalization, and improvement of the dynamic relationship and fact mining problem relative to the Wikipedia domain.
3. The creation of the temporal component as a core measurement to complement content material with semantic data by using user edits.
4. Presentation of cutting-edge automated event validation in cascade to event detection.
5. A new and effective automated event validation approach that outperforms baselines and former state-of-the-art approaches.
6. Considerable boost for precision with a small negative impact on recall of event detection.

In the remainder of the paper, we introduce related works in Section 2. In Section 3, we formalize the problem of dynamic relationships and event discovery and depict a framework for our method. Formalization of the problem for automatic event validation is conducted in Section 4. Sections 5 and 6 describe details of our approach for relationship identification and event detection, while we provide details of our approach for automatic event validation in Section 7. We evaluate our approach in Section 8. Additionally, in this Section, we extensively compare our approach with the well known *Current Event Portal* of Wikipedia (http://en.wikipedia.org/wiki/Portal:Current_events (accessed on the 21 March 2011)) by analyzing the events described manually by Wikipedia users versus the events detected by our model. Finally, we provide our conclusions in Section 10.

2. Related Work

2.1. Temporal Information Retrieval and Event Detection

Inside the area of temporal information retrieval, preceding studies (e.g., [6,7]) insist on chronological measurement in an exclusive ranking fashion in order to enhance the effectiveness for temporal queries regularly with temporal information preprocessing. We stay away from the necessity of indexing or pulling out temporal information from web archives by using Wikipedia. As identified in [8], the hyperlink systems in Wikipedia are a great indicator for the historical impact of human beings. In our work, we recommend the exploitation of the time measurement in an articles' revision in order to discover historical facts.

The hassle of retrieving facts was studied with the use of internet articles as a part of a wider strategy named topic detection and tracking [9], which answers to the following two main questions: "What occurred?" and "What is New?". Two approaches, to the best of our knowledge, exist related to our research: article-based [9,10] and entity-based [11]. In the article-based approach, facts are discovered by using clustering articles based totally on semantics and timestamps. In the entity-based approach, temporal distributions of entities in articles are used to model facts. We discover facts by using dynamic relationships among entities paired at a given time and we devise a complete pipeline as a solution. We chose to apply a retrospective and unsupervised fact detection algorithm by using the users' historical edits collection.

In [3], they propose a method that exploits dynamic relationships in order to describe facts. Proprietary query logs were used to measure the temporal aspect of an entity, observing word matching. The relationships between two entities are solid as to whether their query histories showcase peaks on equal time, arguing that it is a proxy of the concurrent relevance to each entity. In addition to the burst correlation algorithms, empirical experiments had been glaringly laid low with the overall performance of the word matching techniques. With our study, we offer a scientific framework for analyzing the several forms of dynamic entities relationships. We also make use of ontologies consisting of YAGO2 [4] in order to support distinct entity classes and fact domains, as recommended in our previous research [12].

2.2. Automatic Event Validation

The automated facts retrieval was broadly investigated, e.g., in [3,11,13]. However, fact detection is dissimilar from fact check, for the reason that the results of fact retrieval techniques represent the entry for fact check methods. Even though the fact validation task related to a corpus may be similar to a *information Retrieval* assignment, we consider that validating the presence of fact contributors in textual content (e.g., by keyword matching) is not sufficient for checking the incidence of facts in files while establishing mutual connections and chronological conformation.

A first endeavor for automatic fact checking was conducted in [14], where the presence of facts in a textual corpus turned into an evaluation based totally on manual-devised rules. Araki et al. [15] executed historical fact validation as part of *Passage Retrieval*. In contrast to our approach, the authors determine fact validity in phrases of the textual similarity among a bag of words of constant length. Furthermore, the method is analysed exclusively on historic events. In [16], facts are the entry with the scope to append temporal references to documents. Our approach goes beyond this and combines a wide set of features derived from documents and implements a supervised model, as advised in our previous research [17,18].

2.3. Mining from Wikipedia

The survey [19] categorizes and presents different areas relevant to Wikipedia. With respect to our study, there are different research studies on measuring the semantic relatedness of words and entities that use professional-curated taxonomy, such as Wordnet [20] or those that take advantage of the Wikipedia structure [21]. Wikipedia was also exploited as

a rich source for detecting the semantic connections between entities primarily based at the inter-linking structures of Wikipedia articles [22,23] or at the phrasal overlaps extracted from internet articles [24].

New articles attempt retrieving and summarizing historic facts from Wikipedia pages [25]. By studying the developments in article view information as opposed to a Wikipedia pages' edits, the authors in [26] discover concepts with expanded reputation for a specific term, and the authors in [27] suggest a framework to visualize the temporal connections among specific entities. The authors in [2] advance an AI based framework for retrieving and imparting fact-associated information from the Wikipedia edits. A different study suggested constructing a set of associated entities for creating the facts of one entity as reference [28]. In comparison with earlier analysis, we take advantage of the improved Wikipedia editing activity in the contiguity of facts, and we exploit the edit history to become aware of facts and entities showing similar behaviour that might be stricken by the same fact. With our article [29], we primarily endorsed retrieving complex facts of a couple of entities from the Wikipedia edit history in an unsupervised form.

Our research distinguishes itself from the presented works since it focuses on the chronological dynamics of the entities in Wikipedia, i.e., an underlying fact can connect two entities that present low correlation (for instance, *George W. Bush* and *Afghanistan*).

2.3.1. Collaborative Structure in Wikipedia

The collaborative structure of Wikipedia and the availability of the complete edit history has led to research evaluating the degree of controversy of an article [30,31] or the computation of author reputations [32,33]. Different visualizations that use the edit history to extract meaningful information have been proposed, such as wikidashboard [34] or IBM's history flow [35]. Although controversy might be a source for increased edit activity for certain articles, controversy and the author's reputations do not play a direct role in the methods we use. We make use of the edit history, but we do not focus on visualization; in the case that there is a correlation between the increased activity and an event, we identify all the entities involved in the event.

In [30], the authors find controversial articles that attract disagreements between the editors by constructing an edit graph and by using a mutual reinforcement model to calculate disagreement scores for articles. In [31], quantitative indicators for the edit network structure are proposed in order to assess if the authoring community can be partitioned into poles of opinion and if their contributions are balanced. Although it is often the case that the source of increased activity in the edit behaviour is related to controversy, either intrinsic to the entity or having an event as a source, we focus only on exploiting the created content in order to identify relationships that develop because of a common event. In [32], the authors observe that the reputation of the contributors to one article is really relevant to the articles quality. In [33], the edit history is used to compute author reputations and to measure the statistical correlation between the authors' reputation at the time an edit was made and the subsequent lifespan of that edit. The same authors build up on the previous work and design a system (<http://www.wikitrust.net/>) (accessed on the 21 March 2011) that assigns trust values to each word in a Wikipedia article [36], taking into account the article's edit history together with the authentic writer's popularity. We are not evaluating the future stability of text, as event-related updates do not last for long and are replaced with more accurate and fresh information and the authors' reputation do not play a direct role in the methods we use.

2.3.2. Event Summarization in Wikipedia

There are some works that use Wikipedia to extract events and summarize them, such as [37,38], but they do not make use of the edit history. By using facts extracted from just one version of Wikipedia, a visualization tool for events and relationships in Wikipedia is proposed in [39]. The authors of [40] investigate the summarization of edits occurring in a time period as a tag cloud, finding a correlation between the popularity of a topic

and the number of updates, and they make a visualization available. Exploiting the view history rather than the edit history, [27] advances a visualization framework to inspect entity correlations. In [2], the authors propose a framework for identifying fact-related information from the Wikipedia edit history. They only suggest how the presentation of the information can be conducted, but they do not evaluate the proposed summarization methods. Using the edit history, we go a step further and identify all the involved entities and perform an evaluation of our summarization methods.

3. Approach for Event Detection

3.1. Overview

In this article, our purpose is to discover facts from Wikipedia users' edit records. Current studies represent facts by means of actions, as an instance, via an RDF triple of subject-predicate-object [25]. We shape a fact through its collaborating entities: One fact is represented by multiple entities that are connected chronologically. For example, the occasion "Australian Open 2011 women's final" on the 19 January 2011 can be defined by using its players "Li Na" and "Kim Clijsters". This manner of describing facts is independent from a specific language. On the other hand, characterizing the notion of entity connections in order to conduct a fact is not easy. Such relationships need to nicely capture the chronological dynamics of entities in Wikipedia, where information is constantly brought or modified through the years.

The *Explicit Relationship Identification* exploits links among Wikipedia articles to create the connection among their corresponding entities. Each hyperlink that is newly introduced or modified in each article revision shows explicitly a tie among the source and destination entities. For instance, for the duration of the Friday of Anger, the Wikipedia article "National Democratic Party" received many revisions where the hyperlink to the article "Ahmed Shafik" was also edited many times. This discloses a connection between the two entities participating in the rebellion. We explain strategies of this method in Sections 5.1 and 6.1. In the *Implicit Relationships Identification*, we modify the technique suggested in [3] with the scope to represent the entity connections by using burst patterns with spikes driven by means of actual-international activities of the entities [2,26]. In order to prevent the accident of unbiased entities, which burst across the equal chronological period, we enforce that the entities share textual or structure similarities for the duration of the observed time. With the exception of the point-wise mutual information (PMI) [3], we advise different similarity measures for every implicit connection, as mentioned in Section 5.2.

Fact Detection. Once the entity connections are described, we locate facts by using groups of associated entities, each describing a fact. We create a graph where the vertices are the entities and the edges are their connections. This graph is rather dynamic since entity connections are modified over time. In this research study, we advise an adaptive set of rules that manages the chronological dimension, as reported in Section 6.

3.2. Problem Formalization and Data Model

Time is represented as an infinite continuous series $T := [0; +\infty)$ and each time unit/point corresponds to one day and indexed by $i = 1, 2, 3, \dots$. Let E be the entity collection extracted from Wikipedia where each entity e is related to a Wikipedia page, and let τ be the set of time points. For a specific time point i , e is expressed as a textual document $d_e^{(i)}$, which is the version of the page at the latest time point i . With these assumptions, we thus represent the *edit* of e at the time point i as $m_e^{(i)} := d_e^{(i)} - d_e^{(i-1)}$ and the *edit volume* $v_e^{(i)}$ as the number of versions between our two time points $i - 1$ and i .

Dynamic Connections

A dynamic connection is a tuple $r := (e_1, e_2, i)$, where $e_1, e_2 \in E$ are the entities for which r holds and $i \in \tau$ is the time when r is valid. There are two types of dynamic connections: *exact* and *implied*. Strategies identifying the two typologies are described in Section 5.

Events

We define an *event* (fact) v as a tuple $v := (E_v, \tau_v)$, where $E_v \subset E$ is the *characteristic entity set*, i.e., entities that contributed to v and $\tau_v := \{i : i \in \tau, i_{start} \leq i \leq i_{end}\}$ are the *time period* when v happened.

Problem statement

Given an entity set E , a time window $\tau_v \subseteq \tau$ detect all events $v = (E_v, \tau_v)$.

3.3. Workflow

In Figure 1, the workflow of *WikipEvent* is reported. *WikipEvent* is comprised of two phases: *Dynamic Relationship Identification* and *Event Detection*. As entry point, we specify time periods; thus, our architecture retrieves a set of facts together with the involved entities. This representation enables users to understand the retrieved facts (e.g., causes and effects). *Dynamic Relationship Identification* adopts one of the two strategies presented in Section 5.1 (*Exact Connections Retrieval*) and in Section 5.2 (*Implied Connections Retrieval*). The *Exact Connections Retrieval* strategy adopts links in the pages in order to establish the connection among entities, since each link added or edited in one version contains information between the source and destination pages. The modified *Implied Connections Retrieval* strategy comprises two sub-phases. First, we use *Burst Detection* in order to retrieve activity bursts in the edit history, as depicted in Section 5.2. The outcomes are entities pairs possessing bursts in the same chronological interval. In Section 5.2, we present how we utilized a variety of strategies to measure their similarity in the *Entity Similarity* phase. The co-burst graphs for each specific chronological point collect entity pairs.

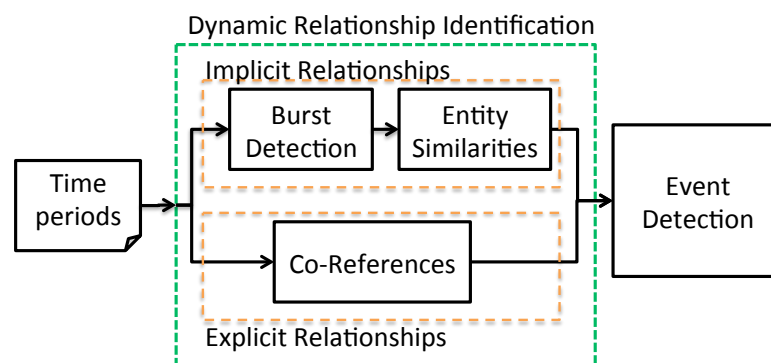


Figure 1. Architecture for retrieving facts and connections among entities.

The second phase, *Event Detection*, creates facts described by representative entities and time intervals. It first constructs a sequence of graphs, each one catching the chronological entity connections. Thus, we incrementally construct the connected components that collect entities that are highly connected in consecutive chronological intervals.

4. Approach for Automatic Event Validation

4.1. Overview

We present the data model in Section 4.2. We define the Event Validation task in Section 4.3, and we show how to exploit the Event Validation task in order to boost the precision of event detection in Section 4.4.

4.2. Problem Formalization and Data Model

Event

An *event* e is a tuple $e := (K_e, t_e^0, t_e^f)$, where $K_e = \{k_e^1, \dots, k_e^{n_e}\}$ is a set of keywords characterizing the contributors, and t_e^0 and t_e^f represent the timespan during which the fact occurred. This follows the guidelines for event definitions presented in previous studies

[3,11,41]; however, other representations as chronological subject-object predicates [4] can also be adjusted to our model with the alignment of the keyword sets.

Document

A document d is represented by its textual content that is subject to scrutiny in order to assess the validity of an event.

Evidence

An event e is said to possess evidence of its presence in a document d with a threshold $\tau, 0 < \tau \leq 1$ iff at least $\tau\%$ of the keywords K_e contribute together in a fact described in d within the fact chronological interval $\{t_e^0, t_e^f\}$.

Pair

Given a fact e and a document d , we associate them within an (*event, document*) pair $p := (e, d)$.

4.3. Event Check

The following two levels of event check exist: document-level and corpus-level. An (e, d) pair may be estimated as *false* if the document does not contain the fact, although the fact might be real and reported in other documents of the corpus. Conversely, a pair that is estimated as *true* may sufficiently demonstrate the validity of a fact at the corpus-level. In general, we aim to check whether a fact is reported in a document or in a corpus.

Document-level. Given a pair p and an evidence threshold τ , we define the *document-level check* as the following function:

$$\gamma_d = V_d(p, \tau) \quad (1)$$

which verifies the evidence γ_d of e in d with threshold τ . The codomain of $V_d(\cdot)$ can vary based on the application requirements. In the case of a binary classification between true and false pairs based on the threshold, $\gamma_d \in \{valid; invalid\}$ is the case; otherwise, one can measure evidence as the percentage of conforming event keywords without considering any threshold and $\gamma_d \in [0, 1]$.

Corpus-level Check. Given a fact e , a set of documents D , and an evidence threshold τ , we define the *corpus-level check* as the following function:

$$\gamma_c = V_c(e, D, \tau) \quad (2)$$

which judges the evidence γ_c of e in the corpus D . Similarly to the document-level check, the codomain of $V_c(\cdot)$ depends on the application requirements.

4.4. Precision Boosting

The automated event check can be considered as a post-processing phase of event retrieval in order to enhance its precision. After the extraction of facts, we consider as true that the automatic event check can improve the general performances by exploiting information that is not available as entry for the fact retrieval. Fact retrieval typically distinguishes facts in a record series primarily on an indistinct input that is described via a huge set of entities [3] or phrases with excessive frequency [11,13]. In comparison, we concentrate on checking their existence in documents.

This is *precision boosting* since the aim is to enhance the precision within the set of detected facts by discarding false ones. However, that process must be completed without affecting recall, i.e., the true events retrieved at the beginning should be kept in the course of the validation. Benefits are proven in Section 9.4.

5. Relationship Identification

As already introduced in Section 3.1, we delve into two strategies to produce dynamic connections.

5.1. Exact Connections Retrieval

In the approach *Co-References*, we provide the entity relationship definition as follows. For each entity e and an edit $m_e^{(i)}$ at time i , if $e_2 \in m_e^{(i)}$, then it indicates that a link exists relative to the Wikipedia page of the entity e_2 in the content of $m_e^{(i)}$. A connection between e_1 and e_2 is created if we have links in both directions.

Definition 1. Given two entities e_1, e_2 and a chronological interval $I = [i, i + \delta]$, an exact dynamic connection between e_1 and e_2 at time point i is a tuple $r_{exp} \in E \times E \times \tau$ such that $r_{exp}(e_1, e_2, i)$ iff $\exists j, k \in [i, i + \delta], e_1 \in m_{e_2}^{(j)}$ and $e_2 \in m_{e_1}^{(k)}$.

Thus, an exact dynamic connection indicates the mutual reference of two entity edits which are made during an overlapping time period. The parameter δ is a delay introduced when adding hyperlinks between entities. For instance, simultaneous with the clear associations of the entities “National Democratic party” and “Ahmed Shafik” with the duration of the 2011 Egypt rebellion, the two mutual references can be viewed at extraordinary (near) time points; for the sake of clarity, the link from the item of “Ahmed Shafi” to “National Democratic party” can be introduced first, and the inverse link can be brought one day after.

5.2. Implied Connections Retrieval

We tailored the Wikipedia domain the approach presented by [3]; in this modified method, we retrieve a connection between entities when their edit histories show bursts within the equal or overlapping chronological interval (co-burst). An entity burst is the time where the number of edits shows an unexpected increase with respect to the historical editing activity of the entity itself. Formally, given an entity e and a chronological interval W , we create a time series $v_e := [v_e^{(i_0)}, \dots, v_e^{(i_f)}]$ by collecting the edit number of entity e at each chronological point $i \in \tau$. A burst b_e is a sequence of chronological points $b_e := [i, i + 1, \dots, i + k]$, for which the edit numbers of e are higher than the edit number detected at previous chronological points: $v_e^{(i-1)} \ll v_e^{(j)}$ and $v_e^{(j)} \gg v_e^{(i+k+1)} \forall j = i, i + 1, \dots, i + k$. Two entities e_1 and e_2 co-burst at time i iff $i \in b_{e_1}$ and $i \in b_{e_2}$.

A co-burst determines entities showing a higher number of edits at the equal chronological point than previously detected. There is the possibility that two entities have co-bursts at the equal chronological interval, even if they contribute to different facts. In order to avoid this, the edits of entities must have sufficient resemblance. The similarity of two entities e_1 and e_2 at time i as $S_{method}(e_1, e_2, i)$ is described using the following similarities:

1. *Textual*: It quantifies how close two entities are at a given time with the aid of evaluating the content material of their corresponding edits. We construct the bags of words $bw_{e_1}^{(i)}$ and $bw_{e_2}^{(i)}$ and use Jaccard index to measure the similarity: $S_s(e_1, e_2, i) := J(bw_{e_1}^{(i)}, bw_{e_2}^{(i)})$;
2. *Entity*: Similarly to *Textual*, but we focus on bag of entities $be_e^{(i)}$ (entities that are linked from the edit): $S_e(e_1, e_2, i) := J(be_{e_1}^{(i)}, be_{e_2}^{(i)})$;
3. *Ancestor*: It quantifies how close two entities are in terms of their semantics. For every entity, we used an ontological knowledge base wherein the entity is registered and extracted all of its ancestors. Given $be_e^{(i)}$, a bag of ancestors $ba_e^{(i)}$ is filled with the ancestors of every entity in $be_e^{(i)}$. We then measure the similarity $S_a(e_1, e_2, i)$ by Jaccard index accordingly;
4. *PMI*: This quantifies how likely entities co-occur in the edits in all other entities. Given $i \in \tau$ and $e_1, e_2 \in E$, we created a graph involving all entities linking to e_1 and e_2 from all edits at i . Let $IN(e)^{(i)}$ be the set of incoming links for e , we estimate the probability of generating e by $p(e) = \frac{IN(e)^{(i)}}{N^{(i)}}$, with $N^{(i)}$, being the total number

of incoming links in the graph at time i . We then calculate the link similarity as $S_{PMI}(e_1, e_2, i) := \log \frac{p(e_1, e_2)}{p(e_1)p(e_2)}$.

6. Event Retrieval

After the formalization of the dynamic entity connections, we now want to retrieve events through their entity sets and the consistent temporal slot. Event retrieval is achieved by using an incremental technique. At each single chronological point $i \in \tau$, we aggregate unique connections into one structure, referred to as *temporal graph*. The temporal graph represents one event's snapshot at the desired time slot (Section 6.2). In order to serve the development of events, we examine entity clusters of adjacent temporal graphs and incrementally merge clusters if a specific criterion is met. The resulting merged set of entities described the fact that spans across sequential days. Our event retrieval is specified hereafter.

6.1. Temporal Graph and Entity Clustering

Definition 2. A temporal graph $G(i)$ at time $i \in \tau$ is an undirected graph (E, P) , where E is an entity set and $P = \{(e_1, e_2) | r(e_1, e_2, j)\}$ is the set of edges defined by dynamic connections at a chronological point $j \in I = [i, i + \delta]$.

Δ reflects the gap of edits among several Wikipedia articles in reaction to one real-global fact. Determined by the dynamic relationships, we have the two following temporal graphs types: *implicit* and *explicit*.

Explicit Temporal Graph Clustering.

Formally, we have a graph $G = (E, P)$, where $P = \{(e_1, e_2) | r_{exp}(e_1, e_2, i)\}$, which we call a *temporal graph*. In this temporal graph, each edge (e_1, e_2) is labelled by the set of all time points where the mutual references are observed: $l(e_1, e_2) = \{i | r_{exp}(e_1, e_2, i)\}$. In an explicit temporal graph, the connection of two nodes, i.e., the edge is defined by the connection $r_{exp}(e_1, e_2, j)$ and indicates the mutual linking structure within the chronological interval I of two Wikipedia pages.

From the temporal graph, we discover cliques C to shape entity clusters which are co-noted from i to $i + \delta$. Every clique $c \in C$ constitutes a fact happening at i . Cliques guarantee the high coherence of the ongoing facts encoded within the group of entities.

Implicit Temporal Graph Clustering.

In an implicit temporal graph, a candidate edge is created between two entities which co-burst at a chronological point $j \in I = [i, i + \delta]$ employing a similarity function.

$$S_{max}(e_1, e_2, I) = \max_{j \in [i, i + \delta]} \{S(e_1, e_2, j)\}$$

We insert an edge (e_1, e_2) if $S_{max}(e_1, e_2, I) \geq \theta$. We build the temporal graph at time i according to these two types of relationships and label each edge (e_1, e_2) by the set $b_{e_1} \cap b_{e_2} \cap I$. Intuitively, θ constitutes the threshold employed to carry out selective pruning, retaining entity pairs with the most similarities. Distinct from an explicit temporal graph, we relax the entity clustering constraints by depicting the facts occurring at i as the connected elements. In fact, two entities e_1 and e_2 that are not immediately associated can nevertheless co-burst via an intermediate entity e' at some point of the interval I within the graph.

6.2. Event Identification

Given a time window $W = [i_{min}, i_{max}]$, this step groups events extracted at each time $i \in W$ to form bigger events that span over continuous chronological points. The algorithm proposed by [3], *Local Temporal Constraint (LTC)*, retrieves facts. At every chronological factor i , *LTC* keeps a group of joined together clusters from the start until i and conglomerates these clusters with those in the temporal graph at time $i + 1$. Clusters are conglomerated by the fact that they share at least one connection. However, it turned

out to be joining a number of clusters wherein the entities across clusters are very loosely associated. We modified the algorithm for the implicit temporal graph: we aggregated the graphs if they share one edge (i.e., the LTC algorithm), while, for explicit temporal graph, we merged clusters c_1 and c_2 if their Jaccard similarity is more than a given threshold γ . Details are presented in Algorithm 1.

Algorithm 1: Algorithm for Entity Cluster Aggregation.

```

Input :  $E, W, strategy$   $\gamma$ -cluster merging threshold
Output:  $C_{ret}$  as entity sets for facts
Set  $C_{ret} = \emptyset$ 
for each  $i \in |W|$  do
  Create temporal graph  $G(i)$ 
   $C_{I_i}$  = clusters in  $G(i)$ 
  for each  $c_k \in C_{I_i}$  do
    merged = False
    for each  $c_j \in C_{I_{i-1}}$  do
      if ( $strategy = explicit$  &  $\frac{|c_k \cap c_j|}{|c_k \cup c_j|} \geq \gamma$ ) or
        ( $strategy = implicit$  &  $|c_k \cap c_j| \geq 1$ ) then
           $c_k = c_k \cup c_j$  (merge the two facts)
          merged = True
        end
      end
    if not merged then
       $C_{ret} = C_{ret} \cup c_k$ 
    end
  end
end
return  $C_{ret}$ 

```

Complexity. Let M be the maximum number of facts within a set C_I . Let n be the maximum number of connections retrievable in C_I . Let \mathcal{T} be the number of intervals, thus the computational cost of LTC is $O((\mathcal{T} - 1)nM^2)$.

7. Automatic Event Validation

7.1. Features

We construct a supervised learning model with the usage of features to predict their validity, i.e., the evidence of events in documents. We describe the features employed in the learning model to predict both the validity and the ambiguity of a given (e, d) pair. Features are extracted from simple text. This makes our technique potentially relevant to any type of corpora. Some features presented hereafter are described by scalar values, while other features are represented as a vector containing the standard deviation, mean, minimum value, and maximum value of the given feature. The term *statistics* will be used hereafter to refer to measures computed in the latter case.

Event Features. The defined event features are grouped and described below. They include the variety of fact keywords and their duration, in addition to the frequency of keywords describing humans, places, corporations, and artefacts.

Document Features. Features are gathered from each document. Our intuition is that characteristics of documents, such as the number and kind of entities appearing in it, along with their distribution in the text can indicate the likelihood that the document contains facts. Thus, we calculate the frequency of words (nouns, verbs, and adjectives) and the named entity types (humans, places, corporations, and artefacts). For each of the words and named entity types, we additionally calculate statistics about their mutual distances and positions. We extract features from chronological information in each document, such as the number of temporal expressions, the statistics of temporal expressions positions,

and the mutual distances in the document. We also consider the length of the document, number of sentences, and words and sentence statistics.

Pair Features. From (e, d) pairs, we can obtain feature pairs. At the beginning, we calculate the proportion of fact keywords exposing as a minimum one match inside the article and information about the variety of matches. Considering that the distance between the matched fact keywords in the document might affect their actual connection, we calculate statistics regarding the mutual distances and positions of matching keywords. The textual matching of fact keywords with words in the document is a necessary but not sufficient condition for the fact to be present in the document, since they might refer to a different timespan than the one relative to the fact. Thus, we gather features with respect to the temporal expressions within the event timespan (*matching dates*, hereafter). We calculate the statistics of both matching and not matching dates about positions and mutual distances. We extract the positions of not matching dates since their presence and proximity to matching dates might be an obstacle to judging the validity of the timespan. In addition, we consider the position of the first date and whether it belongs to the event timespan. In order to estimate the likelihood of fact keywords being mutually connected while referring to a matching date, we calculate features distances. For each matching date and each distinct keyword, the matching word closest to the date is considered.

7.2. Event Validation

Document-level Validation. A Support Vector Machine (SVM) is trained via a 10-fold cross validation over the pairs to predict the validity of new unseen pairs. The label of a pair l_p depends on the scenario that is taken into consideration (Section 4.3): For binary classification, $l_p \in \{true; false\}$; for regression, it is the percentage of fact keywords conforming to the same event in the document and within the event timespan, $l_p \in [0, 1]$.

Corpus-level Validation. Given an event e , a set of candidate documents D , a validity threshold τ , and a document-level validation function $V_d(\cdot)$, our corpus-level validation function $V_c(e, D, \tau)$ in the case of classifications returns *true* if $\exists d \in D : V_d(e, d, \tau) = true$, while in case of regression $V_c(e, D, \tau) = \max_{d \in D} V_d(e, d, \tau)$. The best (e, d) pair evaluated at document-level drives the selections at corpus-level. We selected this strategy in place of others that average the validity values computed at document-level due to the fact that we consider the presence of a fact in an article sufficient to suggest fact validity in other scenarios.

8. Experiments and Evaluations for Event Detection

We wanted to investigate the achievements of the suggested approaches; thus, we executed experiments on the quantitative (Section 8.3) and qualitative (Section 8.4) characteristics of our retrieved facts. For the challenge of retrieving fact structures in Wikipedia, we did not find related work sources containing an exhaustive list of real-world facts. Nevertheless, we discovered several sources attempting to provide a comprehensive list in our research study: (i) the Wikipedia Current Event Portal (cited later as WikiPortal) aggregating and exposing human generated event descriptions; (ii) YAGO2 database [4] representing each fact as a set of entities connections. These approaches are restricted in terms of the number of facts and their degree of abstraction. Furthermore, considering that a complete facts repository was not yet released, computing recall for fact retrieval strategies is infeasible. Hence, we completed a manual evaluation as follows. Retrieved facts have been manually judged by five examiners who had to determine if they were similar to real occasions. For every retrieved fact, the annotators have been asked to check all concerned entities and to pick out a real-global occasion with the aid of inspecting internet-based assets (Wikipedia, legitimate home pages, search engines such as Google, etc.) that best described the co-occurrence of those entities in the fact for the duration of the chronological period. For each set of entities, a label *true* or *false* was assigned for describing the association to a real world occasion.

In conclusion, we executed a comprehensive comparative analysis with the well known WikiPortal by comparing the facts depicted manually by Wikipedia users versus the facts retrieved by our best performing approach (Section 8.5).

8.1. Dataset

Our dataset was created by using English Wikipedia. Due to the fact that Wikipedia also incorporates pages that do not describe entities (e.g., “register of physicists”), we decided to select pages related to entities registered in YAGO2 and those that are a part of the following categories: humans, locations, corporations, and artefacts. In general we retrieved 1,843,665 pages, each related to one entity. We selected a chronological period starting from 18 January 2011 until 9 February 2011, because it covers critical actual-global events which include the Arab Spring rebellion, the Australian Open, and so on. We opted for a relatively short period, since it is easier for humans to match the facts retrieved by our system with real-world occasions by checking the Web. Considering the fact that using days as smallest time unit proved to efficiently seize news-associated facts in both social media and news systems [42], we used day granularity in addition to sampling time. We call the entire dataset containing all pages *Dataset A*. Moreover, we extracted a sample, referred to as *Dataset B*, by deciding on entities that have been actively edited (more than 50 instances) in our temporal window. The idea behind this is that a huge amount of edits can be generated by a fact. Therefore, this sample carries only 3837 pages.

8.2. Implementation Details

Entity Edits Indexing. We used the JWPL Wikipedia Revision Toolkit [43] to save the entire Wikipedia edit records dump and to pick out the edits.

Similarity. In order to retrieve the ancestors of a specific entity, we utilized the YAGO2 knowledge base [4], which is an ontology that was constructed from Wikipedia infoboxes and mixed with Wordnet and GeoNames, to obtain 10 million entities and 120 million records. We observed facts with *subClassOf* and *typeOf* predicates to extract the ancestors of entities. We bounded the retrieval to three tiers, considering the fact that we determined that approaching a higher degree would entail considering many extremely abstract classes. This diminished the discriminating overall performance of the similarity dimension.

Burst Detection and Event Detection. We applied Kleinberg’s algorithm by using the modified version of CShell toolkit (<http://wiki.cns.iu.edu/display/CISHELL/Burst+Detection> (accessed on the 21 March 2011)). We imposed the density scaling to 1.5, the default variety of burst states to 3, and the transition cost to 1.0 (for extra details, read [44]). We noticed that changing the parameters of the burst detection no longer had an effect on the order of overall performance between specific fact retrieval strategies. For the dynamic connections, we imposed the temporal lag parameter δ to 7 days and γ to 0.8, as those values achieved the best outcomes in our experimentation.

8.3. Quantitative Analysis

The purpose of this phase is to numerically assess our strategy below the (i) total quantity of retrieved facts and (ii) the precision, i.e., the proportion of *real* facts. For the parameters’ choice, be aware that the graph created is completely based on the exact method and exposes no weights on its connections. On the other hand, the implied approach generates a weighted graph primarily based at the similarities, and the temporal graph clustering depends on the edge θ for filtering entity pairs of low maximum similarity. We tried different values of θ and observed that decreasing it ended in a bigger wide variety of entity pairs that coalesced right into a low quantity of big facts. These facts having multiple entities could not be recognized as actual facts. Consequently, we used $\theta = 1$ for the following experiments.

We compared methods for the implied connection retrieval strategies described in Section 5.2, known as the following *techniques*: *Textual*, *Entities*, *Ancestors*, and *PMI*, as well as for the exact method as described in Section 5.1 that is referred to as the *Co-References*

method. The outcomes are provided in Tables 1 and 2 where the number of retrieved facts and their precision are reported in the third and fourth columns, respectively. As predicted, we suggest that there are more events in *Dataset A* as in *Dataset B* because of the higher number of entities taken into consideration. The biggest wide variety of retrieved facts is supplied by *Co-references* in each datasets. That is attributed to the parameter-free nature of the exact method, while, with the implied method, a portion of facts are removed by using a threshold. Comparing the methods utilized by the implied approach, *PMI* retrieves more facts than other methods. This is due to the difference in computing the entity similarity $S(e_1, e_2, t)$. *PMI* considers the sets of incoming links that account for relevant feedback to our e_1 and e_2 from all the other entities. This derives more abundant entity pairs and more definite consistent facts, while the other implied strategies generally tend to aggregate entities in bigger but lesser facts. *Textual*, *Entities*, and *Ancestors* calculate $S(e_1, e_2, t)$ starting from the edited contents of two entities at a specific chronological point. A huge amount of content regarding entities, which might no longer be explicitly related to e_1 and e_2 , might be considered, rendering the value of $S(e_1, e_2, t)$ lower. Therefore, using the same value for θ as for the *PMI* creates a lower number of entity pairs and, consequently, of retrieved occasions.

Table 1. Performance on Dataset A.

Strategy	Method	Events	Precision
Exact	Co-References	186	70%
	PMI	124	39%
Implied	Ancestors	33	51%
	Entities	21	62%
	Textual	78	1%

Table 2. Performance on Dataset B.

Strategy	Method	Events	Precision
Exact	Co-References	120	80%
	PMI	80	69%
Implied	Ancestors	18	50%
	Entities	12	60%
	Textual	15	7%

The precision of each setup, i.e., the percentage of authentic retrieved occasions, is summarized in the fourth column of Tables 1 and 2. Considering the various implicit strategy methods, we denoted a clear advantage of using similarities that take semantics into account (*Entities*, *Ancestors*, and *PMI*) over string similarity (*Textual*). *Ancestors* play worse than *Entities* in each datasets, showing that the addition of ancestor entities introduces extra noise rather than clarifying the relationships among the edited entities. *Entities* achieves comparable performances on both datasets. *PMI* achieves better performance in *Dataset B* than compared to the other implicit similarities given that it are exploiting the structure of in-out links among Wikipedia articles. However, *PMI* plays worse on *Dataset A* because of the higher variety of inactive entities taken into consideration, introducing noisy links.

In the end, *Co-References* outperforms all the implied techniques on both datasets. Usually, all approaches carried out better or similar to one another on *Dataset B* versus *Dataset A*, indicating that choosing only the entities which might be edited more frequently improves our strategies. Even though less facts are retrieved in *Dataset B*, the majority of them correspond to actual existence facts.

8.4. Qualitative Analysis

Here, we conduct a qualitative analysis of the facts retrieved in *Dataset A*. We concentrate on a number of facts retrieved by our first-class approach, *Co-References* (Section 8.3). Then, we evaluate some cases wherein our techniques performed poorly and we suggest the possible reasons for the poor performance.

In Table 3, we present some retrieved facts by our best method, *Co-References*, by matching real-world occasions and we relate the entities, the chronological period, and a human readable explanation of the detected fact from web sources for each of them. The Table shows that the behaviour of our methods does not depend on the domain of the entities involved in the events, as we have a good coverage over different fields. In particular, we can observe that sport events are easier to detect because of the highly connected entities and the similar terminology used in the articles of the involved entities. Furthermore, we report the graph structure of two true retrieved facts:

Example 1. We describe the connections to the true occasion called *Friday of Anger* during the Egyptian rebellion in Figure 2a. On 28 January 2011, thousands of people crammed the Egyptian streets to protest against the government. One of the primary demonstrations happened in Cairo. Protests became organized with the help of social media networks and smartphones, and some of the organizers included the April 6 Youth Movement and the National Democratic Party. The aviation minister and previous chief of Air workforce Ahmed Shafiq and Gamal Mubarak are determined as the possible successors of Hosni Mubarak by means of the authorities.

Example 2. The graph structure of some other exceptional actual global occasion that is detected: The declaration of the nominees for the 83rd Academy Awards on 25 January 2011 is shown in Figure 2b. The most crucial node is the 83rd Academy Awards and, as a secondary node, the Academy Award for Best Actor having *True Grit* and *Beautiful* as the connecting nodes, which were been nominated for more classes.

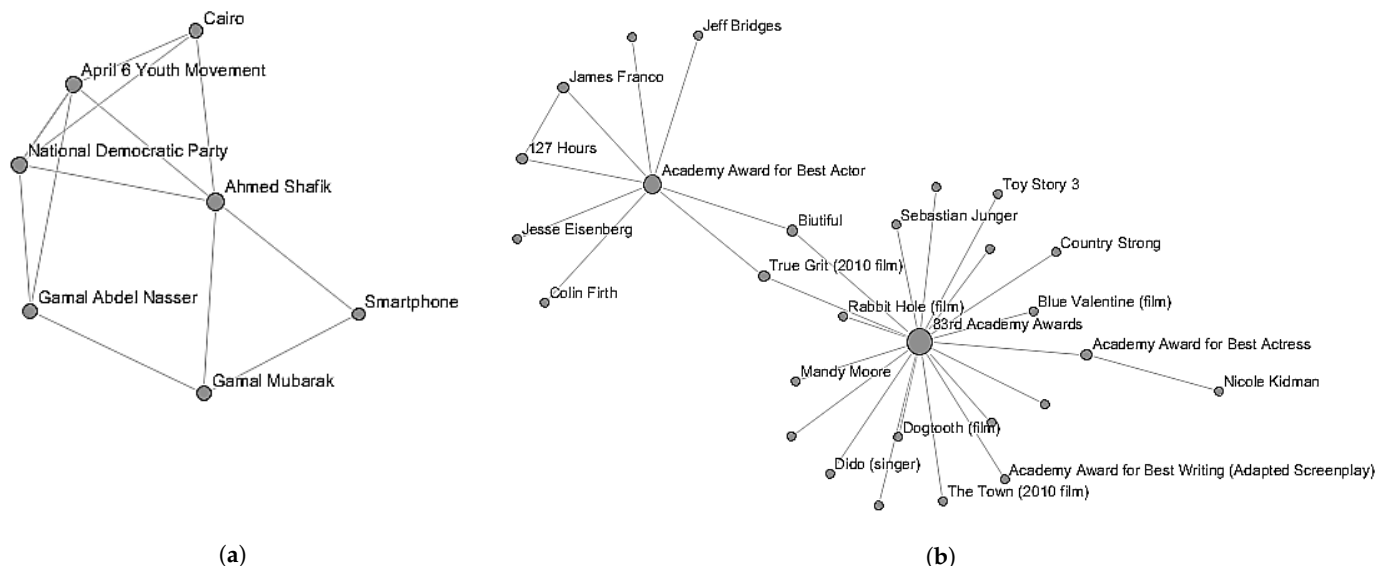


Figure 2. Example of retrieved facts. (a) The Friday of Anger; (b) 83rd Academy Awards Nominations.

We depict some errors of our strategies to retrieve actual-world facts in Table 4, together with the reasons that result in such faulty final results. Depending on the technique, we can identify unique patterns that motivate false positives. The *entity based similarity* generally fails due to updates containing several entities that are not involved in any important event. The usage of the *ancestor-based similarity* can provide false facts because some entities, which can be very comparable and have a wide variety of ancestors, have coincidentally concurrent edit peaks in the temporal window under observation. The *PMI*

shows poor results because of comparable reasons: Entities share several incoming links to the entities contained in the edits conducted on the same day. In the end, the *Co-References* method appears to fail when the reciprocal mentions originate in connections which are unbiased of any fact.

8.5. Comparative Analysis

We examine retrieved facts by means of our approach with facts reported in the WikiPortal inside the identical chronological period. Users in WikiPortal put up a brief summary of a fact in reaction to the incidence of actual-world occurrence and relate the entity mentioned with the corresponding Wikipedia pages. The occasion summary can also be clustered into larger “stories” (which include Egypt rebellion), or it can be prepared in one-of-a-kind class, such as politics, cinema, etc.

We performed the evaluation considering the 130 (70% of 186) true retrieved facts by the *Co-References* approach on *Dataset A*, given that this is the setup that provided the biggest set of authentic facts.

We gathered 561 events from WikiPortal within the specific duration by clustering all occasions in the same story as representing a single fact. In addition, we took into consideration the best event descriptions annotated with a minimum of one entity contained within *Dataset A*, gathering 505 events. These facts can be retrieved via our approach.

Even though an entity can participate in an event without being explicitly annotated within the event description, we assume that the articles of the relevant entities participating in the event are annotated by the users of WikiPortal.

In order to evaluate the overlap between the two fact sets, we classify facts according to the following classes:

- 1. Class Green:** The fact in one set, together with all its contributing entities, is reported in the other set either as a single or as multiple facts.
- 2. Class Yellow:** The fact is partially reported in the other set, i.e., only a subset of its contributing entities is shown in one or more facts in the other set.
- 3. Class Red:** The fact is not present in the other set.

We offer explanations for every class in Table 3 in conjunction with causes of each class selection. We noticed two patterns for green class: (i) One occasion in *Co-References* is spread over several facts in WikiPortal (as an example, the occasion concerning the candidacies for the *Fianna Fail party* is indicated in WikiPortal via many facts and each one specializes in one single candidacy); (ii) one fact in *Co-References* is related to one occasion in WikiPortal. The yellow class typically covers the case where a fact describes a non-stated aspect of a fact in WikiPortal. As an example, for the *Australian Open tennis* event, only the men’s semi-final and very last matches are stated in WikiPortal without mentioning the other ones, which are suggested in *Co-References*. In addition, the *Friday of Anger* (Egyptian rebellion) and the *Academy Awards nominations* are present in WikiPortal, but our retrieved facts are endowed with extra entities that do not appear inside the portal. In the end, the red class aggregates those occasions not mentioned in WikiPortal in any respect, which is similar to the *Royal Rumble wrestling match*.

We observed that 60% of the facts retrieved by *Co-References* are completely or partly reported inside the WikiPortal. For the sake of readability, in Table 5, we provide a number of the events which can be found in WikiPortal, but that our technique was not able to retrieve, in conjunction with an explanation. The main patterns are the following: (i) the fact contains just one entity; (ii) the fact involves entities, which are highly unlikely to reference one another because of their extraordinary roles within the common facts.

In conclusion, *Co-References* and WikiPortal can be considered as complementary strategies for fact retrieval.

Table 3. Co-References: retrieved facts and corresponding real-world occasions.

Entities	Date (2011)	Human Description	Explanation
Class Green			
Brian Cowen, Michel Martin, Mary Hanafin, Mary Coughlan(politician), and Fianna Fáil	From 18 January to 26 January	The Irish PM Brian Cowen announced his stepping down as the leader of the ruling Fianna Fail party and different candidacies for the leadership follow his decision.	The event is globally reported in WikiPortal through different daily events.
Saad Hariri and Najib Mikati	25 January	Supporters of Lebanese caretaker Saad Hariri call for a day of protests following Hezbollah’s support for Najib Mikati as Prime Minister.	The entities participating to the event are all mentioned in an event within WikiPortal.
Class Yellow			
Gamal Abdel Nasser, Ahmed Shafik, Smartphone, Cairo, April 6 Youth Movement, Gamal Mubarak, and National Democratic Party	28 January	In the context of the Egyptian Revolution, the Friday of Anger takes place: tens of thousands filled the streets across Egypt to protest against the government.	Most of the entities appear in WikiPortal within different events. The entities <i>Gamal Mubarak</i> and <i>Gamal Abdel Nasser</i> do not appear.
Li Na and Kim Clijsters	19 January	Australian Open 2011 women’s final: Li Na vs. Kim Clijsters.	The Australian Open is mentioned two times but always focuses on men’s matches. The women’s final is not reported.
James Franco, Colin Firth, Biutiful, True Grit, and 83rd Academy Awards (Figure 2b)	25 January	Announcement of the nominees for the 83rd Academy Awards.	The event is reported in WikiPortal, but few participating entities are mentioned.
Class Red			
Vickie Guerrero, Hornswoggle, Layla El, Dolph Ziggler, Booker T, Professional wrestling, Kane, and Santino Marella	30 January	The 2011 Royal Rumble organized by WWE takes place, involving a lot of wrestlers.	The event and none of its entities are reported in WikiPortal.
Silent Witness, Bruce Forsyth, and Loose Women	26 January	In the context of the 16th National Television Awards presented by Bruce Forsyth, Loose Women and Silent witness are nominated.	The event and none of its entities are reported in WikiPortal.
Catwoman, The Dark Knight, and Bane	19 January	Warner Bros. Pictures announced that Anne Hathaway has been cast as Catwoman and Tom Hardy as Bane in “The Dark Knight Rises”.	The event and none of its entities are reported in WikiPortal.

Table 4. False positive events and possible explanation.

Entities	Approach	Explanation
Alexis Korner, Fleetwood Mac, and Bob Brunning	Co-References	Bob Brunning was a member of Fleetwood Mac (a British-American rock band) and Korner wrote a book about them. They were not involved in any common events in our period.
Alexa Nikolas and Ariana Grande	PMI	Two different persons that only look alike and share some of their own incoming links but not much more. They were not involved in a common event in our period, but their articles might have experienced unrelated edit peaks simultaneously.
Saudi Arabia national football team, Ghana national football team, and Canada national soccer team	Ancestors	The entities have a lot of common ancestors coming from the Sports domain, and all of them had peaks of activity in the same time. However, they were not involved in common events during the studied period.
Tura Satana and Barack Obama	Entities	Tura Satana died, but Barack Obama did not have any connection to her in the period under investigation. Both entities experienced edit peaks and the entities contained in the edits were similar.

Table 5. Examples of events from WikiPortal that were not detected by our method; Co-References.

Event Description	Date (2011)	Explanation
Apple records record profits of \$6 billion as consumers consumed more of its products than was thought (BBC)	18 January	The event involves just one entity.
Chinese President Hu Jintao begins a four-day state visit to the United States.	18 January	It is highly unlikely that the prominent entities have mentioned each other.
Exotic birds are found to have been driven into Britain's back gardens by the extreme cold, as more than half a million people participate in the largest wildlife survey in the world	29 January	It is highly unlikely that the event attracted the attention of the Wikipedia community.
Researchers report that fishing rates in the Arctic are 75 times higher than those reported by the U.N., suggesting future increased exploitation is less possible than previously thought.	4 February	It is highly unlikely that the prominent entities have mentioned each other.

9. Experiments and Evaluations for Automatic Event Detection

9.1. Dataset

The dataset that we took into consideration consists of a set of occasions and candidate pages. We created input pairs starting from this information.

We performed our analysis using the 130 (70% of 186) real facts retrieved with *Co-References* technique on Dataset A, considering that this is the setup that provided the largest set of proper facts. In order to the detected facts, we introduced additional events from a public fact set used by [14]. Overall, the dataset was composed through 258 facts. On this set, fact keywords are titles of Wikipedia articles and the considered chronological period is from the 18 January 2011 to 9 February 2011, as previously noted.

We selected the internet as a source for documents because of its easy accessibility and wide fact coverage. However, any document series can be exploited to construct the ground truth. For every fact, queries were generated by using concatenating the facts keywords along with month and year of the fact timespan. We employed the Bing API to carry out queries and to retrieve the first 20 web-pages for every query. After eliminating duplicates and discarding non-crawlable webpages, we retrieved 7257 pages. Plain text has been extracted from each html page by using BoilerPipe (<http://code.google.com/p/boilerpipe/> (accessed on 21 March 2011)), and the Stanford CoreNLP parser (<http://nlp.stanford.edu/> (accessed on 21 March 2011)) was employed for part-of-speech tagging, named entity and temporal expression retrieval.

Ground Truth. We dissect the task of evaluating whether or not a web-page contains a fact into atomic microtasks, which are then deployed on a premier crowdsourcing platform, CrowdFlower (<http://www.crowdflower.com/> (accessed on 21 March 2011)). For each of the 7257 (e, d) pairs in our dataset, workers had been supplied with a description of the event key phrases and timespan, as well as the report URL. The occasion timespan, which is specified by a start and an end day, was strictly taken into consideration throughout the tasks. The workers were then asked simple questions regarding the occurrence of keywords and their alignment to the stipulated timespan. The task was to report the number of fact keywords conforming to the same fact in the page and within the fact chronological interval (see Figure 3). We observed project design recommendations and hired gold standard questions to discover untrustworthy workers.

Entities: Hosni Mubarak ; Ahmed Shafik ; **Time Interval:** From 18-01-2011 to 07-02-2011

Document URL: <http://www.csmonitor.com/Commentary/Common-Ground/2014/0317/Tunisia-s-model-for-bridging-political-and-social-divides>

How many of the entities conform to the same event within the given range of 18-01-2011 to 07-02-2011? Enter '-1' if the webpage is not available.

Figure 3. Sample questions from the crowdsourced microtask

We collected five different assessments for each pair, ensuing in 36,285 answers in overall (pairwise percentage settlement of 0.7). We collected these judgements to reach the notion of validity for every pair via considering the assessment with the best pairwise percent agreement among workers to signify the validity label (in case of a tie, we took into consideration the label closest to the average of all judgements). From real-valued pair labels l_p , we derived binary validity labels ($l_{p,\tau}$) for pairs (i.e., *true* or *false*) by applying a threshold τ , as discussed in Section 4.3, such that $l_{p,\tau} = true$ iff $l_p > \tau$ and $l_{p,\tau} = false$ otherwise. Those binary labels are used when considering occasion validation as a classification assignment (Section 4.3) and they permit a greater intuitive belief of doc-level validity. In our experiments, we recall three distinctive values for τ : 0.5, 0.65, and 1.0. On the corpus-level, according to techniques described in Section 7.2, 70.9% of the facts suggest a validity value greater than zero. After conducting the same binarization as for

document-level, the shares of *legitimate* facts on the corpus-level are 63.2% ($\tau = 0.5$), 62.7% ($\tau = 0.65$), and 60.4% ($\tau = 1.0$).

9.2. Implementation Details

Evaluation Metrics. We used Cohen’s Kappa between validity labels and the output of our automatic check, both at document and corpus-level. The Cohen’s Kappa (K) defines the level of agreement between two assessments by considering the probability that they randomly agree. For completeness, we calculated the accuracy (ACC) of the approaches. For the reason that authentic validity labels are actual-valued, we also filed the Pearson Correlation Coefficient (r) when predicting them by using regression. Correlation is used as a performance metric to evaluate the ambiguity prediction as well.

Parameter Settings. We trained a SVM via 10-fold cross validation implemented by LibSVM (<http://www.csie.ntu.edu.tw/~cjlin/libsvm/> (accessed on 21 March 2011)) with Gaussian Kernels to classify pairs validity. The settings were tuned to $C = 6.2$, $\gamma = 1.0$ in the case of classification and to $\nu = 0.5$, $\gamma = 0.3$ for regression. We collected the predictions generated during the cross validation.

Baselines. We consider two baselines. *Key-word Matching* (KM) validates pairs by means of counting the proportion of event key phrases found in files. In the case of multi-term keywords, the matching of one term is considered a match for the whole keyword. The second is the method defined in [14], which is indicated as *CF Validation* (CF). The incidence of events in documents is assessed by thinking about the presence of dates in the event timestamps, estimating the regions of text related to these dates, and returning the percentage of event keywords found in these areas. The validation depends on hand-crafted rules, which is a big constraint of the approach.

9.3. Event Validation

We report the outcomes of our fact check strategies in this section.

Document-level. The outcomes of fact check at the document-level, taking into account several validity thresholds ($\tau = 0.5, 0.65, 1.0$) and performance metrics (accuracy ACC , Cohen’s Kappa K , and Pearson Correlation Coefficient r), are indicated in Table 6.

Table 6. Performances of the automatic document-level check.

	$\tau = 0.5$		$\tau = 0.65$		$\tau = 1.0$		-
	ACC	K	ACC	K	ACC	K	r
Baselines							
KM	0.575	0.091	0.575	0.090	0.668	0.191	0.314
CF	0.867	0.401	0.868	0.400	0.870	0.286	0.540
Eventful							
pairs	0.910	0.659	0.909	0.655	0.916	0.605	0.713
all	0.925	0.728	0.923	0.719	0.926	0.680	0.758

Assessment of our approach, named *Eventful*, was conducted by counting on two different models, *pairs* and *all*: The former only regards pair features in the learning, while the latter uses all available features depicted in Section 7.1. Our technique outperforms the baselines under all criteria, showing that combining data from facts and documents by using AI is more powerful than (i) employing mere keyword matching (KM) or (ii) designing check policies that gather connections among fact keywords and temporal conformity to the fact timespan (CF). Specifically, consistent with [45], the values accomplished via our techniques imply a *significant* level of agreement. The enhancement of the *all* model over CF is of 81% for $\tau = 0.5$ and of 138% for $\tau = 1$. Note that the results for CF are lower than the ones reported by Ceroni et al. [14], since here we consider document-level evaluation, while their evaluation was at corpus-level. *Pairs* alone achieves sufficiently better results

than the baselines, but when using support information from facts and documents (*all* model), we reached higher agreement.

We ran outcome comparisons considering several validity thresholds, and we noticed a lowering trend of performances for growing validity thresholds (besides *KM*, resulting in low and noisy results). We ascribe this to the fact that the techniques face obstacles when validating the pairs in all of the fact keywords appearing in a document, however, the most effective subset of them conforms to the same event in the given timespan.

Corpus-level. Table 7 reports the outcomes for corpus-level validation. Our technique outperforms the baselines under all criteria, with *all* achieving the highest performances. Evaluating the outcomes of document-level and corpus-level assessment (Tables 6 and 7) can be determined absolutely that the values within the latter are particularly better. That is because, at the corpus-level, a single legitimate (*e, d*) pair is sufficient to validate the event in spite of the presence of mistakes within the other pairs corresponding to the occasion.

Table 7. Performances of the automatic corpus-level check.

	$\tau = 0.5$		$\tau = 0.65$		$\tau = 1.0$		-
	ACC	K	ACC	K	ACC	K	r
Baselines							
KM	0.663	0.022	0.663	0.022	0.743	0.178	0.410
CF	0.823	0.653	0.827	0.647	0.824	0.601	0.701
Eventful							
pairs	0.904	0.799	0.908	0.802	0.904	0.806	0.862
all	0.876	0.740	0.896	0.782	0.901	0.797	0.845

By analyzing the confusion matrices, we noticed that the learning system increases the share of negative detections (both authentic or false) in order to lessen classification errors. This ensured a lower quantity of false positives at corpus-level regardless of being less correct at the document-level. On the contrary, while the validity threshold was low ($\tau = zero.5$), the learning procedure turned extra confident in classifying legitimate pairs and the ensuing output rate of positive detections became better, resulting in a boom of false positive pairs. This created greater false positive events at the corpus-degree, since a single false positive pair is enough to suggest a false occasion as legitimate.

9.4. Precision Boosting

The effects on the performances of fact retrieval after employing fact check as post-processing step are indicated in Table 8 for several values of τ . P_{det} refers to the precision values (one for each τ) within the set of retrieved facts, consistent with the performances indicated by [3]. The precision and recall in the fact set after applying fact check are reported as P_{val} and R_{val} . Our strategy boosts the precision achieved by event detection up to 89.4% for $\tau = 0.5$ and up to 91.9% for $\tau = 1.0$. The high values of recall (89.9% or more) bring in evidence that most of the true facts in the original set are kept by the check step. Our strategy outperforms *CF* both in precision and especially in recall. With respect to *KM*, the low increase in precision and the high recall represent that performing a check via keyword matching has no effect on the performances of fact retrieval: Most of the facts are assessed as true, since the presence of their keywords in documents is a sufficient condition for the fact check.

Table 8. Effects of applying event validation after event detection.

	$\tau = 0.5$			$\tau = 0.65$			$\tau = 1.0$		
	P_{det}	P_{val}	R_{val}	P_{det}	P_{val}	R_{val}	P_{det}	P_{val}	R_{val}
Baselines									
KM	0.632	0.636	0.993	0.627	0.631	0.995	0.604	0.628	1.000
CF	0.632	0.842	0.773	0.627	0.845	0.771	0.604	0.863	0.618
Eventful									
all	0.632	0.894	0.900	0.627	0.913	0.913	0.604	0.919	0.899

10. Conclusions

We suggest incorporating a temporal component with semantic records to seize dynamic fact structures. By specializing in Wikipedia, we were able to discover historic information, which are facts. No annotated collections have been available and, as a result, we manually judged the overall performance of our approaches with the usage of a set of 1.8 million articles. Over an in depth evaluation, we defined the effectiveness of our suggested strategy. We have shown that an exact connection retrieval strategy achieves better results (maximum precision of 70%) than an implied one. We noticed a further enhancement to 80% in precision when using only actively edited pages. We performed a comparison between the retrieved facts by using our WikiEvent and the ones presented by the WikiPortal within the same chronological interval.

We investigated the hassle of fact checking that is related to a given corpus, offering an effective technique for automatically validating the incidence of events in documents. In order to increase precision, we additionally brought automatic fact check as a post-processing step of fact retrieval, which outperformed baseline and state-of-the-art methods.

In our next steps, we will assess how a current cross-reference of two entities can be combined with an observed low correlation in previous chronological window of the same pair to enhance the quality of fact retrieval. Other dimensions include adding more semantics to the retrieved facts via the means of text analysis. An additional point is considering semantic features to predict validity, as well as by incorporating user feedback via active learning.

Author Contributions: Conceptualization, methodology, formal analysis, investigation, and writing, M.F. and A.C.; software, validation, resources, and data curation A.C.; review, editing and supervision: M.F. Both authors have read and agreed to the published version of the manuscript.

Funding: This work was partially funded by the European Commission in the context of the FP7 ICT projects ForgetIT (grant No. 600826), DURAARK (grant No. 600908), CUBRIK (grant No. 287704), and the ERC Advanced Grant ALEXANDRIA (grant No. 339233).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ebner, M.; Zechner, J.; Holzinger, A. Why is Wikipedia so Successful? Experiences in Establishing the Principles in Higher Education. *J. Univers. Comput. Sci.* **2006**, *6*, 527–536.
2. Georgescu, M.; Kanhabua, N.; Krause, D.; Nejdil, W.; Siersdorfer, S. Extracting Event-Related Information from Article Updates in Wikipedia. In Proceedings of the 35th European conference on Advances in Information Retrieval, Moscow, Russia, 24–27 March 2013; pp. 254–266.
3. Das Sarma, A.; Jain, A.; Yu, C. Dynamic Relationship and Event Discovery. In Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, Hong Kong, China, 9–12 February 2011.
4. Hoffart, J.; Suchanek, F.; Berberich, K.; Weikum, G. YAGO2: A Spatially and Temporally Enhanced Knowledge Base from Wikipedia. *Artif. Intell.* **2012**, *194*, 28–61. [[CrossRef](#)]

5. Ceroni, A.; Gadiraju, U.; Fisichella, M. JustEvents: A Crowdsourced Corpus for Event Validation with Strict Temporal Constraints. In Proceedings of the Information Retrieval—39th European Conference on Research, ECIR 2017, Aberdeen, UK, 8–13 April 2017; Volume 10193, pp. 484–492. [\[CrossRef\]](#)
6. Alonso, O.; Strötgen, J.; Baeza-Yates, R.; Gertz, M. Temporal Information Retrieval: Challenges and Opportunities. *Twaw* **2011**, *11*, 1–8.
7. Efron, M.; Golovchinsky, G. Estimation Methods for Ranking Recent Information. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, 24–27 July 2011; pp. 495–504.
8. Takahashi, Y.; Ohshima, H.; Yamamoto, M.; Iwasaki, H.; Oyama, S.; Tanaka, K. Evaluating significance of historical entities based on tempo-spatial impacts analysis using Wikipedia link structure. In Proceedings of the 22nd ACM Conference on Hypertext and Hypermedia, Eindhoven, The Netherlands, 6–9 June 2011. [\[CrossRef\]](#)
9. Allan, J.; Papka, R.; Lavrenko, V. On-Line New Event Detection and Tracking. In Proceedings of the SIGIR98: 21st Annual ACM/SIGIR International Conference on Research and Development in Information Retrieval, Melbourne, Australia, 24–28 August 1998; pp. 37–45.
10. Fisichella, M.; Stewart, A.; Denecke, K.; Nejd, W. Unsupervised public health event detection for epidemic intelligence. In Proceedings of the 19th ACM International Conference on Information and Knowledge Management, Toronto, ON, Canada, 26–30 October 2010; pp. 1881–1884.
11. He, Q.; Chang, K.; Lim, E.P. Analyzing Feature Trajectories for Event Detection. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, 23–27 July 2007; pp. 207–214.
12. Tran, T.; Ceroni, A.; Georgescu, M.; Naini, K.D.; Fisichella, M. WikipEvent: Leveraging Wikipedia Edit History for Event Detection. In *Lecture Notes in Computer Science, Proceedings of the Web Information Systems Engineering—WISE 2014—15th International Conference, Thessaloniki, Greece, 12–14 October 2014; Part II*; Benatallah, B., Bestavros, A., Manolopoulos, Y., Vakali, A., Zhang, Y., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8787, pp. 90–108. [\[CrossRef\]](#)
13. Fung, G.P.C.; Yu, J.X.; Yu, P.S.; Lu, H. Parameter Free Bursty Events Detection in Text Streams. In Proceedings of the 31st International Conference on Very Large Data Bases, Trondheim, Norway, 30 August–2 September 2005.
14. Ceroni, A.; Fisichella, M. Towards an Entity-Based Automatic Event Validation. In *Lecture Notes in Computer Science, Proceedings of the Information Retrieval—36th European Conference on IR Research, ECIR 2014, Amsterdam, The Netherlands, 13–16 April 2014*; de Rijke, M., Kenter, T., de Vries, A.P., Zhai, C., de Jong, F., Radinsky, K., Hofmann, K., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8416, pp. 605–611. [\[CrossRef\]](#)
15. Araki, J.; Callan, J. An Annotation Similarity Model in Passage Ranking for Historical Fact Validation. In Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, Queensland, Australia, 6–11 July 2014.
16. Hovy, D.; Fan, J.; Gliozzo, A.; Patwardhan, S.; Welty, C. When Did That Happen? Linking Events and Relations to Timestamps. In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, 23 April 2012.
17. Ceroni, A.; Gadiraju, U.K.; Fisichella, M. Improving Event Detection by Automatically Assessing Validity of Event Occurrence in Text. In Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, 19–23 October 2015; Bailey, J., Moffat, A., Aggarwal, C.C., de Rijke, M., Kumar, R., Murdock, V., Sellis, T.K., Yu, J.X., Eds.; ACM: New York, NY, USA, 2015; pp. 1815–1818. [\[CrossRef\]](#)
18. Ceroni, A.; Gadiraju, U.; Matschke, J.; Wingert, S.; Fisichella, M. Where the Event Lies: Predicting Event Occurrence in Textual Documents. In Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2016, Pisa, Italy, 17–21 July 2016; Perego, R., Sebastiani, F., Aslam, J.A., Ruthven, I., Zobel, J., Eds.; ACM: New York, NY, USA, 2016; pp. 1157–1160. [\[CrossRef\]](#)
19. Medelyan, O.; Milne, D.; Legg, C.; Witten, I.H. Mining meaning from Wikipedia. *Int. J. Hum. Comput. Stud.* **2009**, *67*, 716–754. [\[CrossRef\]](#)
20. Budanitsky, A.; Hirst, G. Evaluating WordNet-based Measures of Lexical Semantic Relatedness. *Comput. Linguist.* **2006**, *32*, 13–47. [\[CrossRef\]](#)
21. Hassan, S.; Mihalcea, R. Semantic Relatedness Using Salient Semantic Analysis. In Proceedings of the AAAI 2011, San Francisco, CA, USA, August 7–11 2011.
22. Ponzetto, S.P.; Strube, M. Knowledge derived from wikipedia for computing semantic relatedness. *J. Artif. Int. Res.* **2007**, *30*, 181–212. [\[CrossRef\]](#)
23. Milne, D.; Witten, I.H. *An Effective, Low-Cost Measure of Semantic Relatedness Obtained from Wikipedia Links*; AAAI: Menlo Park, CA, USA, 2008.
24. Hoffart, J.; Seufert, S.; Nguyen, D.B.; Theobald, M.; Weikum, G. KORE: Keyphrase Overlap Relatedness for Entity Disambiguation. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM 2012, Maui, HI, USA, 29 October–2 November 2012; pp. 545–554.
25. Kuzey, E.; Weikum, G. Extraction of temporal facts and events from wikipedia. In *Proceedings of the 2nd Temporal Web Analytics Workshop*; ACM: New York, NY, USA, 2012; pp. 25–32.

26. Ciglan, M.; Nørnvåg, K. WikiPop: Personalized Event Detection System Based on Wikipedia Page View Statistics. In Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, ON, Canada, 26–30 October 2010; pp. 1931–1932.
27. Peetz, M.H.; Meij, E.; de Rijke, M. OpenGeist: Insight in the Stream of Page Views on Wikipedia. In Proceedings of the SIGIR Workshop on Time-aware Information Access, Portland, OR, USA, 16 August 2012.
28. Tran, T.; Elbassuoni, S.; Preda, N.; Weikum, G. *CATE: Context-Aware Timeline for Entity Illustration*; ACM: New York, NY, USA, 2011; pp. 269–272.
29. Ceroni, A.; Georgescu, M.; Gadiraju, U.; Naini, K.D.; Fisichella, M. Information Evolution in Wikipedia. In *OpenSym '14, Proceedings of the International Symposium on Open Collaboration*; Association for Computing Machinery: New York, NY, USA, 2014; pp. 1–10. [\[CrossRef\]](#)
30. Vuong, B.Q.; Lim, E.P.; Sun, A.; Le, M.T.; Lauw, H.W. On Ranking Controversies in Wikipedia: Models and Evaluation. In Proceedings of the WSDM '08, 2008 International Conference on Web Search and Data Mining, Palo Alto, CA, USA, 11–12 February 2008.
31. Brandes, U.; Kenis, P.; Lerner, J.; van Raaij, D. Network analysis of collaboration structure in Wikipedia. In Proceedings of the 18th International Conference on World Wide Web, New York, NY, USA, 20–24 April 2009.
32. Stein, K.; Hess, C. Does it matter who contributes: a study on featured articles in the German Wikipedia. In Proceedings of the Eighteenth Conference on Hypertext and Hypermedia, Manchester, UK, 10–12 September 2007.
33. Adler, B.T.; de Alfaro, L. A content-driven reputation system for the Wikipedia. In Proceedings of the 16th International Conference on World Wide Web, New York, NY, USA, 8–12 May 2007.
34. Suh, B.; Chi, E.H.; Kittur, A.; Pendleton, B.A. Lifting the Veil: Improving Accountability and Social Transparency in Wikipedia with Wikidashboard. In Proceedings of the Twenty-Sixth Annual CHI Conference, Florence, Italy, 5–10 April 2008.
35. Viégas, F.B.; Wattenberg, M.; Dave, K. Studying Cooperation and Conflict between Authors with History Flow Visualizations. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '04, Vienna, Austria, 24–29 April 2004.
36. Adler, B.T.; Chatterjee, K.; de Alfaro, L.; Faella, M.; Pye, I.; Raman, V. Assigning Trust to Wikipedia Content. In Proceedings of the WikiSym08: 2008 International Symposium on Wikis, Porto, Portugal, 8–10 September 2008.
37. Hienert, D.; Luciano, F. Extraction of Historical Events from Wikipedia. *arXiv* **2012**, arXiv:1205.4138.
38. Hienert, D.; Wegener, D.; Paulheim, H. Automatic Classification and Relationship Extraction for Multi-Lingual and Multi-Granular Events from Wikipedia. In Proceedings of the Detection, Representation, and Exploitation of Events in the Semantic Web, Boston, MA, USA, 12 November 2012.
39. Sipoš, R.; Bhole, A.; Fortuna, B.; Grobelnik, M.; Mladenić, D. Demo: HistoryViz—Visualizing Events and Relations Extracted from Wikipedia. In *The Semantic Web: Research and Applications*; Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvönen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5554. [\[CrossRef\]](#)
40. Nunes, S.; Ribeiro, C.; David, G. WikiChanges: Exposing Wikipedia Revision Activity. In Proceedings of the 4th International Symposium on Wikis, Porto, Portugal, 8–10 September 2008.
41. McMinn, A.J.; Moshfeghi, Y.; Jose, J.M. Building a Large-Scale Corpus for Evaluating Event Detection on Twitter. In Proceedings of the 22nd ACM International Conference on Information and Knowledge Management San Francisco, CA, USA, 27 October–1 November 2013.
42. Bandari, R.; Asur, S.; Huberman, B.A. The Pulse of News in Social Media: Forecasting Popularity. *arXiv* **2012**, arXiv:1202.0332.
43. Ferschke, O.; Zesch, T.; Gurevych, I. Wikipedia Revision Toolkit: Efficiently Accessing Wikipedia's Edit History. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, USA, 19–24 June 2011; pp. 97–102.
44. Kleinberg, J. Bursty and Hierarchical Structure in Streams. *Data Min. Knowl. Discov.* **2002**, 91–101. [\[CrossRef\]](#)
45. Landis, J.R.; Koch, G.G. The Measurement of Observer Agreement for Categorical Data. *Biometrics* **1977**, 33, 159–174. [\[CrossRef\]](#) [\[PubMed\]](#)