# The Knowledge Awakens: Keeping Knowledge Bases Fresh with Emerging Entities

Johannes Hoffart
Max Planck Institute for Informatics
jhoffart@mpi-inf.mpg.de

Dragan Milchevski
Max Planck Institute for Informatics
dmilchev@mpi-inf.mpg.de

Gerhard Weikum
Max Planck Institute for Informatics
weikum@mpi-inf.mpg.de

Avishek Anand
L3S Research Center
anand@l3s.de

Jaspreet Singh
L3S Research Center
singh@l3s.de

## ABSTRACT

Entity search over news, social media and the Web allows users to precisely retrieve concise information about specific people, organizations, movies and their characters, and other kinds of entities. This expressive search mode builds on two major assets: 1) a knowledge base (KB) that contains the entities of interest and 2) entity markup in the documents of interest derived by automatic disambiguation of entity names (NED) and linking names to the KB. These prerequisites are not easily available, though, in the important case when a user is interested in a newly emerging entity (EE) such as new movies, new songs, etc. Automatic methods for detecting and canonicalizing EEs are not nearly at the same level as the NED methods for prominent entities that have rich descriptions in the KB.

To overcome this major limitation, we have developed an approach and prototype system that allows searching for EEs in a user-friendly manner. The approach leverages the human in the loop by prompting for user feedback on candidate entities and on characteristic keyphrases for EEs. For convenience and low burden on users, this process is supported by the automatic harvesting of tentative keyphrases. Our demo system shows this interactive process and its high usability.

## Keywords

knowledge base life cycle, knowledge base curation, entity disambiguation, emerging entities, human in the loop

## 1. MOTIVATION AND INTRODUCTION

**Motivation.** Connecting texts to knowledge bases (KBs) by linking names to the KB's canonical entities such as people, organizations, or movies and their characters, is a fundamental first step for a broad range of applications. Beyond the tasks of language understanding, question answering, and information extraction, one key application that has recently emerged is the use of entities in information retrieval. Searching by entities as well as the possibility to search for classes of entities has been prominently showcased by Google's knowledge graph, but also by recent academic projects like Broccoli [1] or STICS [4]. However, entity-based search crucially depends on the user-queried entities being present in the KB. This is problematic with emerging entities (EEs), i. e. entities that are completely new or are just gaining popularity, as it often takes considerable time for EEs to be added to them [8].

Imagine for example a Star Wars fan, who — after seeing the latest movie — wants to know what others think of her favorite character `Finn` by looking at Social Media. Searching for just the string "Finn" will turn up a lot of uninteresting results about other Finns, e. g. the English cricketer `Steven Finn` or Tom Sawyer's friend `Huckleberry Finn`. This is exactly the use case that entity-based search addresses, solving the ambiguity and allowing users to search with crisp entities — given that the entity is present in the KB of course. However, the new movie character Finn might not (yet) be in the KB.

**Problem.** The problem is to quickly identify EEs and gather a sufficiently crisp description suitable for users to understand the entity and for linking further texts to the new entity, allowing documents to be indexed with the EE and making it searchable. Such descriptions are one of the fundamental building blocks for entity linking methods [10]. They are used for computing the textual similarity of an (ambiguous) name in a text and an entity in a KB. However, automated approaches to keep knowledge bases fresh with EEs (including automatically harvested descriptions) are not accurate enough [3, 7] to work in a completely unsupervised manner.

**Human in the Loop.** The most promising way to achieve human-like quality when adding entities is with with the help of the fan herself. However, even such manual curation must be well supported by the system to avoid putting undue burden on the user.

A straightforward way to obtain the description would be to ask the user for phrases that are salient and descriptive for the EE to be added. However, this would soon become boring for the user and result in poor keyphrases. Additionally, checking if the entity already exists in the KB is cumbersome. When users lose attention and care, there is a high risk of adding duplicate entities.

**Solution.** Our idea to keep the user engaged and motivated is to present her with entities-in-context (EICs, see

*WWW'16 Companion,* April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4144-8/16/04.
http://dx.doi.org/10.1145/2872518.2890537.

... The *Force Awakens*' premiere. *British actor John Boyega*, 23, has one of the film's *lead roles* as `Finn`, a *redeemed stormtrooper*. ...

... untitled *Episode VIII*, which is due out in 2017. *Londoner Boyega*, who auditioned for seven months to play `Finn`, said: "For a *guy from south-east Peckham*, I think I did alright." ...

... franchise has been thoroughly hilarious. *New characters* `Finn` and *Rey* play off each other wonderfully with *quick retorts* and *sparks of chemistry*. ...

**Figure 1: Entities-in-Context (EICs) for the Star Wars character Finn (keyphrases are *emphasized*)**

Figure 1), i. e. snippets of text containing the entity name and some context, which she merely has to accept or reject. This is a low-overhead activity, and a lot of people do this in everyday applications, for example, to tag faces in photographs (Apple's iPhoto comes to mind) or to find matching partners (Tinder). From accepted EICs we can automatically distill keyphrases to create an on-the-fly description for the EE. Additionally, to reduce the likelihood for adding duplicates to the KB, the user is presented existing entities based on the current EE names and description.

Our demonstration implements this idea in an interactive Web application that allows users to add EEs to a KB using news articles as source for generating EICs.

## 2. ADDING EMERGING ENTITIES

The key requirement for adding new entities is that the representation should be suitable for disambiguating the entity in new texts. There is a large and growing body of work on entity disambiguation [10], and many methods using different features have been created over the past years. Crucial features are the importance of an entity with respect to the KB (and sometimes the *mention*), the *coherence* between entities in a single text, and the *textual description* of an entity. In principle, almost all features can be mined from an EIC. In this paper we focus on keyphrases, i. e. the textual description of entities, as the central feature.

### 2.1 Architecture

We assume a sufficiently large collection of documents $\mathcal{D}$ that serve as a repository from which to choose EICs. The architecture to add new entities connects several components, which are depicted in Figure 2. Components requiring user feedback are marked with the shape of a head. The goal is to minimize the effort on the user side by presenting as few EICs as possible, while at the same time achieving a high likelihood that the EIC is about the entity in question. The detailed process is as follows:

1. The user provides a set of names $\mathcal{N}$ and an initial description in the form of keyphrases $\mathcal{K}$. These keyphrases can be very few, maybe only one or two highly salient ones, or even the name alone in case it is not too ambiguous. In any case, the keyphrases are only needed to get the actual iterative process started where the user never has to actively provide keyphrases anymore.

2. Candidate EICs $\mathcal{D}_{\text{cand}} \subset \mathcal{D}$ are retrieved by querying for all the strings in $\mathcal{N}$ using the open source ElasticSearch.[1]

3. While the entity is not added to the KB:
   - Each EIC is scored based on the overlap between the EIC's context and $\mathcal{K}$ (see Section 2.2). The score thus computed for each $d \in \mathcal{D}_{\text{cand}}$ serves as a ranking. Note that the scoring with small initial $\mathcal{K}$ can go wrong, which is exactly why user feedback is necessary.
   - Until the user accepts an EIC, i. e. stating that the entity shown corresponds to the one to be added, $d_i$ are presented in descending score order.
   - The accepted EIC is mined for keyphrases, which are added to $\mathcal{K}$. The user has the option to reject single keyphrases here. Once the user is satisfied, the contextual overlap score is recomputed based on the updated $\mathcal{K}$.

When the entity is finally saved to the knowledge base, additional statistics like co-occurrence counts between entities and keyphrases can be mined from the accepted EICs to e. g. compute keyphrase weights, further improving the disambiguation quality.

A more detailed look is needed for three important aspects here, namely how to score EICs (Section 2.2), how to harvest keyphrases for an accepted EIC (Section 2.3) and how to decide when enough context has been gathered so that an entity can be finally added to the KB (Section 2.4).

### 2.2 Scoring Entities-in-Context

In principle, any disambiguation method that makes use of textual description provided in $\mathcal{K}$ can be used. In this demo we are using AIDA [5], which computes the contextual similarity based on keyphrases associated with the entity. Thus, the same way users describe entities is used for the actual disambiguation. AIDA's contextual similarity scoring is computed as follows:

$$\text{score}(e) = \sum_{k \in \mathcal{K}(e)} \frac{\#\ \textit{token matches}}{\textit{length cover}(k)} \left( \frac{\sum_{t \in cover} \phi(t)}{\sum_{t \in k} \phi(t)} \right)^2,$$

where cover is the shortest span of tokens in the EIC covering the maximum number keyphrase $k$'s tokens, and $\phi$ is the *normalized pointwise mutual information* between the entity and each $t$.

### 2.3 Harvesting Entity Keyphrases

Once an EIC has been approved by the user, keyphrases can be mined in several ways. As the user is involved and can give feedback on wrongly extracted keyphrases, simple methods should suffice. For example, regular expressions over part-of-speech patterns can be used to harvest

[1] https://www.elastic.co

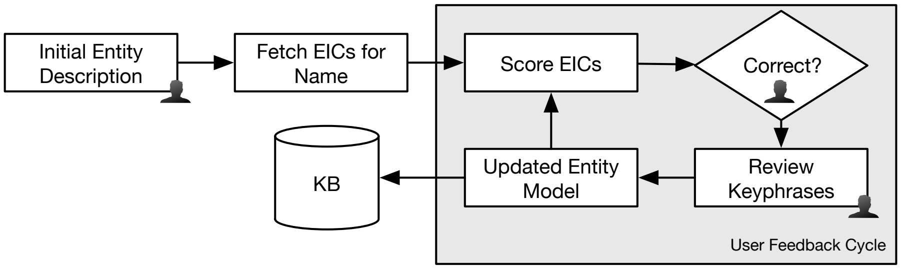


**Figure 2: Interactive Keyphrase Mining with User Feedback**

keyphrase candidates. These patterns would serve as a filter to include useful phrases. In practice, noun phrases that include proper nouns (i. e. names or parts of names) and technical terms have been shown to be useful [3].

## 2.4 Growing the Knowledge Base

One key question that remains is when a sufficiently rich description of the new entity is reached so that it can be added to the KB. This depends on three criteria:

1. The searcher, reading the selected keyphrases, should be able to recognize the entity.
2. From the perspective of the disambiguation method, the keyphrases need to be different enough from all confusable entities, i. e. entities sharing at last one name with the new entity.
3. The entity to be added must not already be present in the KB.

Criterion (1) can be as simple as the user deciding that the description is sufficient, e. g. by having to confirm the addition of the entity to the KB, highlighting that she should be able to understand the entity simply by looking at the keyphrases. Criterion (2) can be assessed by using the very same method that is used to compute the contextual similarity when doing the actual disambiguation, assuming the user feedback on the EICs as ground truth. If the disambiguation method, with the current state of context, is able to provide the same yes/no decisions for each EIC as the user did, the entity can be added to the KB. Criterion (3) can be satisfied by continuously presenting potentially confusable entities to the user during the process, which reduces the likelihood of adding duplicate entities.

# 3. DEMO SCENARIO

For our demo, we use a continuously updated collection of more than 3 million news articles gathered from over 300 sources since June 2013. These articles contain almost 60 million mentions of about 600,000 distinct entities from Wikipedia.

The process of interactively adding entities can be initiated with very little effort, as the screenshot in Figure 3 shows. The process goes as follows:

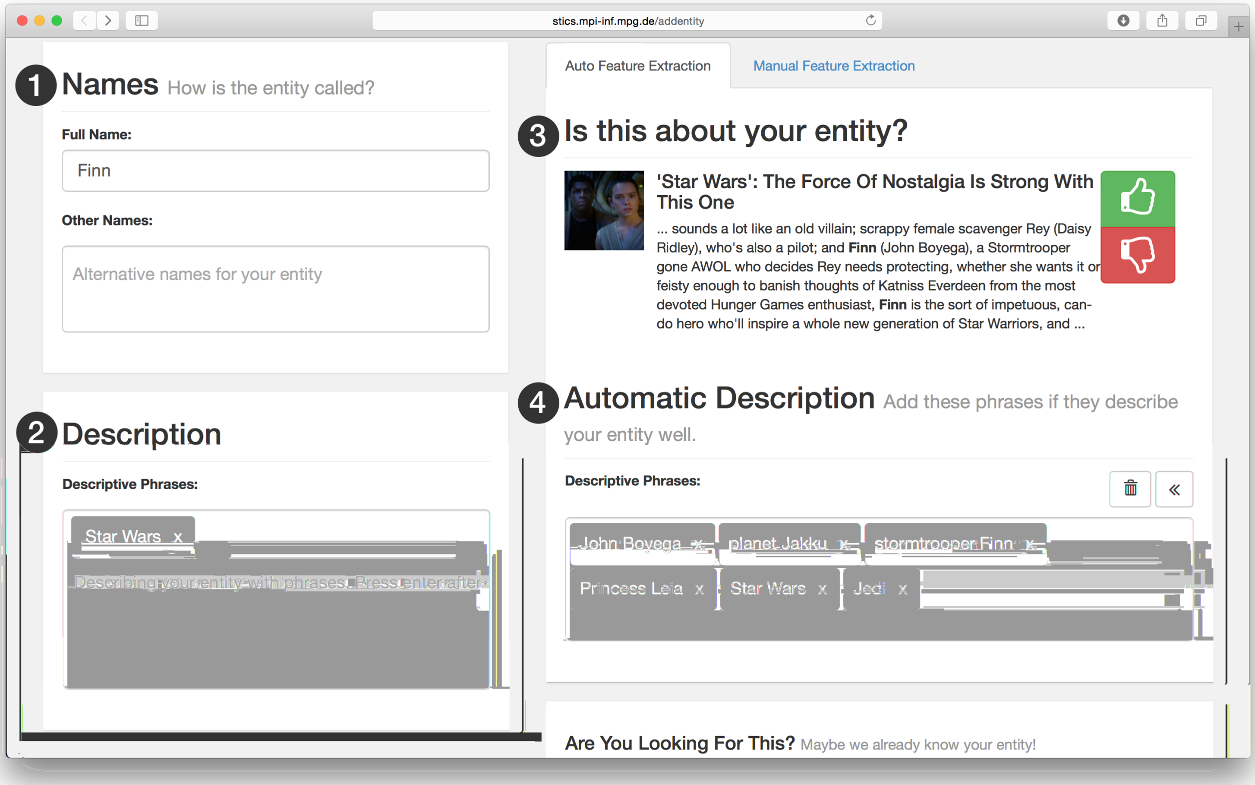
1. The user enters the canonical name in the *Names* panel, e. g. "Finn". For entities that are known by multiple names, e. g. `Star Wars VII`, additional names like the subtitle "The Force Awakens" can be added.
2. The initial *Description* can be minimal, in the case of `Finn` it suffices to add "Star Wars" as initial keyphrase. In the case where the correct Finn is very prominent in recent news, as is often the case with emerging entities, the description can be left completely empty and the user can start with the next step immediately.
3. The user can continue with the more convenient entity-in-context example in the *Is this about your entity?* pane, which shows a snippet where `Finn` is mentioned and our entity disambiguation system linked it to the entity the user wants to enter.
4. If the user accepts, keyphrases will be mined from the text surrounding the snippet and presented to the user for verification below the snippet. The user can accept each individual keyphrase by clicking on it, or add them all by clicking the ≪ button, after which the keyphrases will be added to the main *Description* pane (2).

This process continues until there is a sufficiently rich description of the entity, which allows it to be added to the KB. The addition of duplicate entities to the knowledge base is avoided by showing possible entities that fit the description in the *Are Your Looking For This?* pane, see Figure 4. If the user actually wanted to add the cricketer `Steven Finn`, this would be immediately visible to her.

# 4. RELATED WORK

There is ample work on automatically identifying new or emerging entities. This task has been part of the TAC Knowledge Base Population track [6] since its inception. Here, mentions referring to entities that are not part of the knowledge base should be identified and clustered by meaning. These clusters could in principle be added to a KB as new entities, but the precision of about 75% [7] is still not nearly high enough to do this without human supervision. Other works have focused on extending existing entities with new keyphrases mined from a collection [9, 3].

A natural application where users need entities going beyond Wikipedia-based knowledge bases is entity-based search. Here, the goal is to retrieve documents linked to KBs by querying for contained entities or categories [2, 1, 4]. To the best of our knowledge, our work is the first to propose a retrieval-assisted *manual entity addition* for high quality entity representations for emerging and long-tail entities.



1 Names How is the entity called?

Full Name: Finn

Other Names: Alternative names for your entity

2 Description

Descriptive Phrases: Star Wars x

Describing your entity with phrases. Press enter after

3 Is this about your entity?

'Star Wars': The Force Of Nostalgia Is Strong With This One

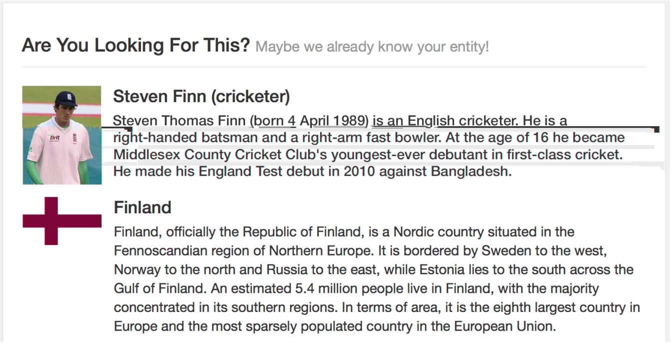
... sounds a lot like an old villain; scrappy female scavenger Rey (Daisy Ridley), who's also a pilot; and **Finn** (John Boyega), a Stormtrooper gone AWOL who decides Rey needs protecting, whether she wants it or feisty enough to banish thoughts of Katniss Everdeen from the most devoted Hunger Games enthusiast, **Finn** is the sort of impetuous, can-do hero who'll inspire a whole new generation of Star Warriors, and ...

4 Automatic Description Add these phrases if they describe your entity well.

Descriptive Phrases: John Boyega x planet Jakku x stormtrooper Finn x Princess Leia x Star Wars x Jedi x

Are You Looking For This? Maybe we already know your entity!

**Figure 3: Interactively adding Finn to the knowledge base**


Are You Looking For This? Maybe we already know your entity!

Steven Finn (cricketer)

Steven Thomas Finn (born 4 April 1989) is an English cricketer. He is a right-handed batsman and a right-arm fast bowler. At the age of 16 he became Middlesex County Cricket Club's youngest-ever debutant in first-class cricket. He made his England Test debut in 2010 against Bangladesh.

Finland

Finland, officially the Republic of Finland, is a Nordic country situated in the Fennoscandian region of Northern Europe. It is bordered by Sweden to the west, Norway to the north and Russia to the east, while Estonia lies to the south across the Gulf of Finland. An estimated 5.4 million people live in Finland, with the majority concentrated in its southern regions. In terms of area, it is the eighth largest country in Europe and the most sparsely populated country in the European Union.

**Figure 4: Alternative entities named Finn**

## 5. REFERENCES

[1] H. Bast, F. Bäurle, B. Buchhold, and E. Haußmann. Semantic Full-Text Search with Broccoli. In *SIGIR 2014*, 2014.

[2] J. Dalton, L. Dietz, and J. Allan. Entity query feature expansion using knowledge base links. In *SIGIR 2014*, 2014.

[3] J. Hoffart, Y. Altun, and G. Weikum. Discovering Emerging Entities with Ambiguous Names. In *WWW 2014*, 2014.

[4] J. Hoffart, D. Milchevski, and G. Weikum. STICS: Searching with Strings, Things, and Cats. *SIGIR 2014*, 2014.

[5] J. Hoffart, M. A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, and G. Weikum. Robust Disambiguation of Named Entities in Text. In *EMNLP 2011*, 2011.

[6] H. Ji, R. Grishman, and H. T. Dang. Overview of the TAC2011 Knowledge Base Population Track. In *Text Analysis Conference*, 2011.

[7] H. Ji, J. Nothman, B. Hachey, and F. Radu. Overview of TAC-KBP2015 Tri-lingual Entity Discovery and Linking. In *Text Analysis Conference*, 2015.

[8] B. Keegan, D. Gergle, and N. Contractor. Hot Off the Wiki: Structures and Dynamics of Wikipedia's Coverage of Breaking News Events. *American Behavioral Scientist*, 57(5), 2013.

[9] Y. Li, C. Wang, F. Han, J. Han, D. Roth, and X. Yan. Mining Evidences for Named Entity Disambiguation. In *KDD 2013*, 2013.

[10] W. Shen, J. Wang, and J. Han. Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions. *IEEE Trans. Knowl. Data Eng.*, 27(2), 2015.