Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence

YAGO2: A Spatially and Temporally Enhanced Knowledge Base from Wikipedia (Extended Abstract)*

Johannes Hoffart **Fabian M. Suchanek** **Klaus Berberich** **Gerhard Weikum**
Max Planck Institute for Informatics
Saabücken, Germany
{jhoffart,suchanek,kberberi,weikum}@mpi-inf.mpg.de

Abstract

We present YAGO2, an extension of the YAGO knowledge base, in which entities, facts, and events are anchored in both time and space. YAGO2 is built automatically from Wikipedia, GeoNames, and WordNet. It contains 447 million facts about 9.8 million entities. Human evaluation confirmed an accuracy of 95% of the facts in YAGO2. In this paper, we present the extraction methodology and the integration of the spatio-temporal dimension.

1 Introduction

Comprehensive knowledge bases in machine-readable representations have been an elusive goal of AI for decades. Seminal projects such as Cyc [Lenat, 1995] and WordNet [Fellbaum, 1998] manually compiled common sense and lexical (word-sense) knowledge, yielding high-quality repositories on intensional knowledge: general concepts, semantic classes, and relationships like hyponymy (subclass-of) and meronymy (part-of). These early forms of knowledge bases contain logical statements that songwriters are musicians, that musicians are humans and that they cannot be any other species, or that Canada is part of North America and belongs to the British Commonwealth. However, they do not know that Bob Dylan and Leonard Cohen are songwriters, that Cohen is born in Montreal, that Montreal is a Canadian city, or that both Dylan and Cohen have won the Grammy Award. Early resources like the original Cyc and WordNet lacked extensional knowledge about individual entities of this world and their relationships (or had only very sparse coverage of such facts).

In the last few years, the great success of Wikipedia and algorithmic advances in information extraction have revived interest in large-scale knowledge bases and enabled new approaches that could overcome the prior limitations [Hovy *et al.*, 2013]. Notable endeavors of this kind include DBpedia [Auer *et al.*, 2007], KnowItAll [Banko *et al.*, 2007], WikiTaxonomy [Ponzetto and Strube, 2007], and YAGO [Suchanek *et al.*, 2007; Hoffart *et al.*, 2013], and meanwhile there are also commercial services such as *freebase.com*. These contain many millions of individual entities, their mappings into semantic classes, and relationships between entities.

However, current state-of-the-art knowledge bases are mostly blind to the temporal dimension. They may store birth dates and death dates of people, but they are unaware of the fact that this creates a time span that demarcates the person's existence and her achievements in life. They are also largely unaware of the temporal properties of events. For example, they may store that a certain person is the president of a certain country, but presidents of countries or CEOs of companies change. Even capitals of countries or spouses are not necessarily forever. Therefore, it is crucial to capture the time periods during which facts of this kind actually happened. A similar problem of insufficient scope can be observed for the spatial dimension. Purely entity-centric representations know locations and their located-in relations, but they do not consistently attach a geographical location to events and entities. The geographical location is a crucial property not just of physical entities such as countries, mountains, or rivers, but also of organization headquarters, or events such as battles, fairs, or people's births. All of these entities have a spatial dimension.

This paper presents an endeavor to create an ontology anchored in the spatial and temporal dimension: YAGO2. As the name suggests, this is a new edition of the YAGO knowledge base. However, in contrast to the original YAGO, the methodology for building YAGO2 (and also maintaining it) is systematically designed top-down with the goal of integrating entity-relationship-oriented facts with the spatial and temporal dimensions. To this end, we have developed an extensible approach to fact extraction from Wikipedia and other sources, and we have tapped on specific inputs that contribute to the goal of enhancing facts with spatio-temporal scope. The most obvious application of such a spatio-temporal knowledge base is that it becomes possible to ask for distances between places, such as organization headquarters and cities (already possible today), or even between places of events (mostly not supported today). The time-awareness would allow asking temporal queries, such as "Give me all songs that Leonard Cohen wrote after Suzanne". In addition, YAGO2 incorporates carefully selected keywords and keyphrases that characterize entities; these are automatically gathered from the contexts where facts are extracted. As no knowledge

*This paper is an extended abstract of the AI journal publication [Hoffart et al., 2013]

base can ever be complete, the contextual annotations further enhance the capabilities for querying and interactive exploration.

The result is YAGO2, available at http://www.yago-knowledge.org. It contains more than 447 million facts for 9.8 million entities (if GeoNames entities are included). Without GeoNames entities, it still contains 124 million facts for 2.6 million entities, extracted from Wikipedia and WordNet. Both facts and entities are properly placed on their temporal and geographical dimension, thus making YAGO2 a truly time and space aware ontology. More than 30 million facts are associated with their occurrence time, and more than 17 million with the location of their occurrence. The time of existence is known for 47% of all entities, the location for 30%. Sampling-based manual assessment shows that YAGO2 has a precision (i.e., absence of false positives) of 95 percent (with statistical significance tests).

2 Extensible Extraction Architecture

The YAGO2 extraction architecture is based on declarative rules, which reduces the hard-wired extraction code to a method that interprets the rules. The rules take the form of subject-predicate-object-triples, so that they are basically additional YAGO2 facts. There are different types of rules.

Factual rules are simply additional facts for the YAGO2 knowledge base. They are declarative translations of all the manually defined exceptions and facts that the previous YAGO code contained. These include the definitions of all relations, their domains and ranges, and the definition of the classes that make up the YAGO2 hierarchy of literal types (`yagoInteger` etc.). Each literal type comes with a regular expression that can be used to check whether a string is part of the lexical space of the type.

Implication rules say that if certain facts appear in the knowledge base, then another fact shall be added. Thus, implication rules serve to deduce new knowledge from the existing knowledge. For example, one of the implication rules states that if a relation is a sub-property of another relation, then all instances of the first relation are also instances of the second relation.

Replacement rules say that if a part of the source text matches a specified regular expression, a certain string should replace it. This takes care of interpreting micro-formats, cleaning up HTML tags, and normalizing numbers.

Extraction rules say that if a part of the source text matches a specified regular expression, a sequence of facts shall be generated. These rules apply primarily to patterns found in the Wikipedia infoboxes, but also to Wikipedia categories, article titles, and even other regular elements in the source such as headings, links, or references.

This architecture for extraction rules is highly versatile and easily extensible. It allows accommodating new infoboxes, new exceptions, new fact types, and new preprocessing by simply modifying the text files of rules.

3 Temporal Dimension

We choose a pragmatic approach to give YAGO2 a temporal dimension, because we can derive the temporal properties of objects from the data we have in the knowledge base.

We consider temporal information for both entities and facts:

- *Entities* are assigned a time span to denote their existence in time. For example, Elvis Presley is associated with `1935-01-08` as his birthdate and `1977-08-16` as his time of death.
- *Facts* are assigned a time point if they are instantaneous events, or a time span if they have an extended duration with known begin and end. For example, the fact `BobDylan created BlondeOnBlonde` is associated with the time point `1966-05-16` (the release date of this album).

Thus, YAGO2 assigns begin and/or end of time spans to all entities, to all facts, and to all events, if they have a known start point or a known end point. If no such time points can be inferred from the knowledge base, it does not attempt any assignment. Thereby, YAGO2 chooses a conservative approach, leaving some time-dependent entities without a time scope, but never assigning an ill-defined time.

Many entities come into existence at a certain point of time and cease to exist at another point of time. People, for example, are born and die. Countries are created and dissolved. Some entities come into existence, but never cease to exist. This applies to abstract creations such as pieces of music, scientific theories, or literature works.

Instead of manually considering each and every entity type as to whether time spans make sense or not, we focused on the following four major entity types with the relations that indicate their time span: **people** (with `wasBornOnDate` and `diedOnDate`), **groups** (such as music bands, football clubs, universities, or companies; with the relations `wasCreatedOnDate` and `wasDestroyedOnDate`), **artifacts** (such as buildings, paintings, books, music songs, or albums; with the relations `wasCreatedOnDate` and `wasDestroyedOnDate`), and **events** (such as sports competitions like Olympics, or named epochs like the "German autumn"; with the relations `startedOnDate` and `endedOnDate`). Note that the entities are already captured in richly populated types within YAGO2, covering three quarters of all entities (not including the GeoNames locations).

Rather than dealing with each of the above four types in a separate manner, we unify these cases by introducing two generic *entity-time relations*: `startsExistingOnDate` and `endsExistingOnDate`. Both are an instance of the general `yagoRelation` and hold between an entity and an instance of `yagoDate`. They define the temporal start point and end point of an entity, respectively. Certain relations are sub-properties of the generic ones, like `wasBornOnDate` or `diedOnDate`, defining existence timespans for time-dependent entities. Our infrastructure generates existence times for all entities where such data can be deduced from Wikipedia.

Facts, too, can have a temporal dimension. The fact `BarackObama holdsPoliticalPosition PresidentOfTheUnitedStates` denotes an epoch from the time Obama was elected until either another president is elected or Obama resigns. When we can extract

time information for these kinds of facts from Wikipedia, we associate it as *occurrence time*: the time span when the fact occurred. To capture this knowledge, we introduce two new relations, `occursSince` and `occursUntil`, each with a (reified) fact and an instance of `yagoDate` as arguments. For example, if the above fact had the fact id `#1`, we would indicate its time by `#1 occursSince 2009-01-20`.

The YAGO2 extractors can find occurrence times of facts from the Wikipedia infoboxes. For example, awards are often mentioned with the year they were awarded. Spouses are often mentioned with the date of marriage and divorce. Our extractors can detect these annotations and attach the corresponding `occursSince` and `occursUntil` facts directly to the target fact.

In some cases, the entities that appear in a fact may indicate the occurrence time of the fact. For example, for `BobDylan wasBornIn Duluth`, it seems most natural to use Dylan's birth date as the fact's occurrence time. For `BobDylan created BlondeOnBlonde`, it should be the creation time of the object.

The principle for handling these situations is to use rules that propagate the begin or end of *an entity's existence time* to the *occurrence time of a fact*, where the entity occurs as a subject or object. To avoid a large number of rules for many specific situations, we categorize relations into several major cases. Each of these has an ontological interpretation, and each can be handled by a straightforward implication rule.

4 Spatial Dimension

All physical objects have a location in space. For YAGO2, we are concerned with entities that have a permanent spatial extent on Earth – for example countries, cities, mountains, and rivers. Geographical coordinates, consisting of latitude and longitude, can describe the position of a geo-entity. YAGO2 only knows about coordinates, not polygons, so even locations that have a physical extent are represented by a single geo-coordinate pair. As we extract these coordinates from Wikipedia, the assignment of coordinates to larger geo-entities follows the rules given there.

Harvesting Geo-Entities YAGO2 harvests geo-entities from two sources. The first source is Wikipedia, which contains a large number of cities, regions, mountains, lakes, etc, many of which come with geographical coordinates.

However, not all geo-entities in Wikipedia are annotated with geographical coordinates. Furthermore, there are many more geo-entities than are known to Wikipedia. Therefore, we tap into an even richer source of freely available geographical data: GeoNames (http://www.geonames.org), which contains data on more than 7 million locations. GeoNames classifies locations in a flat category structure, and each location is assigned only one class, e. g. `Berlin` is a "capital of a political entity". To integrate this data in YAGO2, we match the individual geo-entities that exist both in Wikipedia and GeoNames, so that we do not duplicate theses entities when extracting them from the respective repositories, as well as the assigned class. Individuals are matched when their names are the same. In the case of ambiguity, they are matched if their geographical coordinates are close enough. The GeoNames classes are matched with a handpicked subset of YAGO classes with geographical meaning. Each GeoNames class is mapped to the class in the subset with the highest token overlap of their glosses.

This matching process augments YAGO2 with over 7 million geo-entities and over 320 million new facts from GeoNames, in particular adding geographical coordinates that could not be extracted from Wikipedia, which renders more entities accessible by spatial queries.

Assigning a Location We deal with the spatial dimension in a manner similar to the way we deal with time, as described in Section 3: we assign a location to both entities and facts wherever this is ontologically reasonable and wherever this can be deduced from the data. The location of facts and entities is given by a geo-entity. For example, the location of the *Summer of Love* is San Francisco.

Our knowledge base contains such spatial data for the following types of entities: **events** that took place at a specific location, such as sports competitions, **groups or organizations** that have a venue, such as the campus of a university, and **artifacts** that are physically located somewhere, like the Mona Lisa in the Louvre.

Not only entities have a spatial dimension, this is also the case for facts. For example, the fact that Leonard Cohen was born in 1934 happened in his city of birth, Montreal. Naturally, not all facts have a spatial dimension: for example, schema-level facts such as `subclassOf` or identifier relations such as `hasISBN` have no location on Earth. Again, the key to a semantically clean treatment of the spatial dimension of facts lies in the relations.

Some facts occur in a place that is indicated by their subject or object. For example, the fact that Jimi Hendrix was born in Seattle happened in Seattle. Examples of such relations are `wasBornIn`, `diedIn`, `worksAt`, and `participatedIn`.

Some relations occur in tandem: One relation determines the location of the other. For example, `wasBornOnDate` defines the time of the corresponding `wasBornIn` fact, and the latter defines the location of the former. The first relation specifies the time of the event while the second specifies the location. Other examples include the pairs `diedOnDate`/`diedIn` and `happenedOnDate`/`happenedIn`.

5 Textual Dimension

YAGO2 does not just contain a time and a location for facts and entities, but also meta information about the entities. This includes non-ontological data from Wikipedia as well as multilingual data.

Non-Ontological Data from Wikipedia For each entity, YAGO2 contains *contextual information*. This context is gathered by our extractors from Wikipedia: each anchor text, each category name, and each citation title occurring in an article is added as keywords for the entity. These keywords are useful for searching knowledge in YAGO2, e.g. to make

factual queries more specific or to increase the coverage of entity-specific queries when essential facts are missing, but also for other tasks such as entity disambiguation [Hoffart *et al.*, 2011].

Multilingual Information For individual entities, we extract multilingual translations from inter-language links in Wikipedia articles. This allows us to refer to and query for YAGO2 individuals in foreign languages. YAGO2 represents these non-English entity names through reified facts. For example, we have the reified fact `#1: BattleAtWaterloo isCalled SchlachtBeiWaterloo` with the associated fact `#1 inLanguage German`.

This technique works for the individuals in YAGO2, but not for the classes, because the taxonomy of YAGO2 is taken from WordNet, which is in English. To fill this gap, we integrate the Universal WordNet (UWN) [de Melo and Weikum, 2009] into YAGO2. UWN maps words and word senses of WordNet to their proper translations and counterparts in other languages. For example, the French word "école" is mapped to its English translation "school" at the word level, but only to specific meanings of school at the word-sense level, as the French word does never denote, e.g., a school of fish or a school of thought. UWN contains about 1.5 million translations and sense assignments for 800,000 words in over 200 languages at a precision of over 90% [de Melo and Weikum, 2009]. Overall, this gives us multilingual names for most entities and classes in YAGO2.

6 Factual Evaluation and Numbers

Our main goal for the construction of the YAGO2 ontology was near-human accuracy. This section presents an evaluation of the knowledge base quality. In the ideal case, we would compare the data in YAGO2 to some prior ground truth. Such ground truth, however, is not available for YAGO2, so we had to rely on human judgment.

Our evaluation concerns only the base facts of YAGO2, not the facts derived by implication rules. It only considers the "semantic" relations (such as `wasBornOnDate`) and not the "technical" relations (such as `hasWikipediaURL`). In our methodology [Suchanek *et al.*, 2007], human judges are presented with randomly selected facts, for which they have to assess the correctness. Since the judges might not have enough knowledge to assess each fact, the Wikipedia page from which the fact was extracted is presented next to the fact. Thus, the judges evaluate the correctness of YAGO2 with respect to the content of Wikipedia. We do not assess the factual correctness of Wikipedia itself. We used the Wikipedia dump from 2010-08-17 for the YAGO2 evaluation.

26 judges participated in our evaluation. Over the course of a week, they evaluated a total number of 5,864 facts. This gave us an accuracy value for each sample. We estimate the accuracy of the entire pool by the fraction of samples that were assessed as true, and we compute a Wilson confidence interval [Brown *et al.*, 2001] for each evaluated relation. We evaluated until the confidence interval was smaller than $\pm 5\%$. This ensures that the results are statistically significant.

Relation	#Facts	Accuracy
`actedIn`	126,636	$97.36\% \pm 2.64\%$
`created`	225,563	$98.04\% \pm 1.96\%$
`graduatedFrom`	15,583	$96.84\% \pm 3.16\%$
`hasGender`	804,747	$94.58\% \pm 5.07\%$
`influences`	18,653	$95.28\% \pm 4.42\%$
`isMarriedTo`	27,708	$96.89\% \pm 3.11\%$
`subclassOf`	367,0409	$93.42\% \pm 2.67\%$
`type`	8,414,398	$97.68\% \pm 1.83\%$

Table 1: Evaluation of selected relations

Table 1 describes the results for some of the important non-temporal, non-spatial relations, Table 2 shows the results for temporal and spatial ones. Results for all relations are available at http://www.yago-knowledge.org.

Relation	#Facts	Accuracy
`diedIn`	28,834	$97.91\% \pm 2.09\%$
`diedOnDate`	315,659	$97.68\% \pm 2.32\%$
`happenedIn`	11,694	$96.50\% \pm 3.50\%$
`happenedOnDate`	27,563	$97.86\% \pm 2.14\%$
`isLocatedIn`	436,184	$96.50\% \pm 3.50\%$
`livesIn`	20,882	$96.79\% \pm 3.21\%$
`wasBornIn`	90,181	$96.36\% \pm 3.64\%$
`wasBornOnDate`	686,053	$96.79\% \pm 3.21\%$
`wasCreatedOnDate`	507,733	$97.43\% \pm 2.41\%$
`wasDestroyedOnDate`	23,617	$96.15\% \pm 3.61\%$

Table 2: Evaluation of temporal and spatial relations

The evaluation shows the very high accuracy of our extractors. The vast majority of facts, 97.80%, were judged correct. This results in an overall Wilson center (weighted average over all relations) of 95.40% with a width of $\pm 3.69\%$. The crucial taxonomic relations are `type` (categorizing the individuals into classes) and `subclassOf` (linking classes).

7 Conclusions

We have developed a methodology for enriching large knowledge bases of entity-relationship-oriented facts along the dimensions of time and space, and we have demonstrated the practical viability of this approach by the YAGO2 ontology comprising more than 447 million facts of near-human quality. We believe that such spatio-temporal knowledge is a crucial asset for many applications including entity linkage across independent sources (e.g., in the Linked-Data cloud [Bizer *et al.*, 2009]) and semantic search. Along the latter lines, we think that the combined availability of ontological facts and contextual keywords makes querying and knowledge discovery much more convenient and effective.

Since the work on YAGO2, we have developed the knowledge base further. The new version is called YAGO2s [Biega *et al.*, 2013]. We have improved the overall architecture, and can now provide the data for download in different thematic datasets. Furthermore, the data format of YAGO is now fully RDF compliant.

References

[Auer *et al.*, 2007] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary G. Ives. DBpedia: A Nucleus for a Web of Open Data. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea*, pages 722–735, 2007.

[Banko *et al.*, 2007] Michele Banko, Michael J. Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. Open Information Extraction from the Web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI 2007, Hyderabad, India*, pages 2670–2676, 2007.

[Biega *et al.*, 2013] Joanna Biega, Erdal Kuzey, and Fabian Suchanek. Inside YAGO2s: A Transparent Information Extraction Architecture. In *Demo at WWW 2013*, 2013.

[Bizer *et al.*, 2009] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. DBpedia - A Crystallization Point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154 – 165, 2009.

[Brown *et al.*, 2001] Lawrence D. Brown, Tony T. Cai, and Anirban Dasgupta. Interval Estimation for a Binomial Proportion. *Statistical Science*, 16(2):101–133, May 2001.

[de Melo and Weikum, 2009] Gerard de Melo and Gerhard Weikum. Towards a Universal Wordnet by Learning from Combined Evidence. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM 2009, Hong Kong, China*, pages 513–522, 2009.

[Fellbaum, 1998] Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[Hoffart *et al.*, 2011] Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust Disambiguation of Named Entities in Text. In *Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, Edinburgh, Scotland, 2011*, pages 782–792, 2011.

[Hoffart *et al.*, 2013] Johannes Hoffart, Fabian M Suchanek, Klaus Berberich, and Gerhard Weikum. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artificial Intelligence*, 194:28–61, 2013.

[Hovy *et al.*, 2013] Eduard Hovy, Roberto Navigli, and Simone Paolo Ponzetto. Collaboratively built semi-structured content and Artificial Intelligence: The story so far. *Artificial Intelligence*, 194:2–27, 2013.

[Lenat, 1995] Douglas B. Lenat. CYC: A Large-Scale Investment in Knowledge Infrastructure. *Commun. ACM*, 38(11):32–38, 1995.

[Ponzetto and Strube, 2007] Simone Paolo Ponzetto and Michael Strube. Deriving a Large-Scale Taxonomy from Wikipedia. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence, AAAI 2007, Vancouver, British Columbia, Canada, 2007*, pages 1440–1445, 2007.

[Suchanek *et al.*, 2007] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. YAGO: A Core of Semantic Knowledge. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Canada*, pages 697–706, 2007.