# On Using Wikipedia to Build Knowledge Bases for Information Extraction by Text Segmentation

Elton Serra, Eli Cortez, Altigran S. da Silva, Edleno S. de Moura

Universidade Federal do Amazonas, Brazil
{eltonsrc,eccv,alti,edleno}@dcc.ufam.edu.br

**Abstract.** We propose a strategy for automatically obtaining datasets from Wikipedia to support unsupervised Information Extraction by Text Segmentation (IETS) methods. Despite the importance of preexisting datasets to unsupervised IETS methods, there has been no proper discussion in the literature on how such datasets can be effectively obtained or built. We report experiments in which three state-of-the-art unsupervised IETS methods use datasets obtained according to our proposed strategy under several configurations, involving IETS tasks on three different domains. The results suggest that our strategy is valid and effective, and that IETS methods can achieve a very good performance if the datasets generated have a reasonable number of representative values on the domain of the data to be extracted.

Categories and Subject Descriptors: H.2 [**Database Management**]: Miscellaneous; I.2.6 [**Artificial Intelligence**]: Learning

General Terms: Algorithms, Performance, Experimentation

Keywords: Data Management, Information Extraction, Knowledge Bases

## 1. INTRODUCTION

The dominant approach in information extraction by text segmentation (IETS) is the deployment of probabilistic frameworks such as Hidden Markov Models (HMM) [Borkar et al. 2001] or Conditional Random Fields models (CRF) [Lafferty et al. 2001] to automatically learn a model for extracting data related to an application domain. These methods usually require training data consisting of a set of representative segmented and labeled input strings. Currently, methods based on CRF are the state-of-the-art, outperforming HMM-based methods in experimental evaluations presented in the literature [Sarawagi 2008; Zhao et al. 2008].

As an example of the information extraction task performed by a typical IETS method, consider the input ad *"Regent Square $228,900 1028 Mifflin Ave.; 6 Bedrooms; 2 Bathrooms. 412-638-7273"*. A correct extraction process over this string would generate a structured record such as:

This work is partially supported by INWeb (MCT/CNPq grant 57.3871/2008-6), by project MinGroup (CNPq grant 575553/2008-1), by UOL Bolsa Pesquisa program (grant 20110212103900), by the authors' individual grants and scholarships from CNPq, CAPES and FAPEAM and was developed in cooperation with Nhemu Technologies.

⟨neighborhood,"*Regent Square*"⟩,
⟨price,"*$228,900*"⟩,
⟨number,"*1028*"⟩,
⟨street,"*Mifflin Ave.;*"⟩,
⟨bedrooms,"*6 Bedrooms;*"⟩,
⟨bathrooms,"*2 Bathrooms.*"⟩,
⟨phone,"*412-638-7273*"⟩

To alleviate the need for manually labeled training data, recent IETS methods [Mansuri and Sarawagi 2006; Agichtein and Ganti 2004] rely on preexisting datasets (dictionaries, knowledge bases, references tables) from which content-based features (e.g., vocabulary, value range, format) can be learned. Such features are known to be very effective as state features in sequential models, such as Conditional Random Fields (CRF) [Lafferty et al. 2001]. Besides saving user effort, using preexisting datasets also makes the process of learning content-based features independent from the input texts. In addition, it has been recently shown that input-independent content-based features can be used to bootstrap the learning of input-dependent structure-related features, which are used as transition features in sequential models. Thus, as these datasets allow for the unsupervised learning of both content and structure related features, a number of fully unsupervised IETS methods have recently emerged [Agichtein and Ganti 2004; Zhao et al. 2008; Cortez et al. 2010; Cortez et al. 2011].

Despite the importance of such preexisting datasets to unsupervised IETS methods, there has been no proper discussion in the literature on how such datasets can be effectively obtained or built. In fact, experiments with IETS methods reported in the literature have been carried out using datasets obtained in different ways (e.g., personal files, information extraction benchmarks, etc.), but no principled methods for generating them have been proposed so far.

In this article, we take a first step in this direction. Specifically, we propose a strategy that uses Wikipedia for automatically generating datasets to support unsupervised IETS methods such as Unsupervised CRF (U-CRF) [Zhao et al. 2008], ONDUX [Cortez et al. 2010] and JUDIE [Cortez et al. 2011]. The common point in all of them is that, as stated above, to perform their extraction tasks they strictly rely on preexisting datasets. We also show that the proposed strategy is feasible for supporting real IETS tasks and that, according to our experiments, the generated datasets lead to high-quality extraction results.

Wikipedia* currently contains high volumes information structured in the form of *articles*, *categories*, *infoboxes* and *citations*. More importantly, this information covers a huge diversity of topics and domains. This has been attracting the attention of researches towards its use as a source of domain knowledge for various data management, data mining and information retrieval methods for the Web [Arguello et al. 2009; Wang and Domeniconi 2008; Fuxman et al. 2009; Overell et al. 2009; Hu et al. 2008; Hu et al. 2009; Weikum and Theobald 2010], and, in particular, for Named Entity Recognition [Ratinov and Roth 2009; Weld et al. 2009]. Besides, the usage of structured information to created (develop) extraction methods [Lange et al. 2010; DeRose et al. 2007; Cafarella et al. 2008] is being explored in the information extraction realm.

In our case, we consider the typical IETS scenario, in which semi-structured data records are to be extracted by identifying attribute values occurring in continuous text, such as bibliographic citations, product descriptions, classified ads, etc. We propose a strategy for automatically generating, from an XML dump of Wikipedia, representative datasets on the domain of the attributes involved in a given IETS task.

To evaluate this strategy, we performed experiments in which datasets generated with it are used

*http://en.wikipedia.org

with state-of-the-art unsupervised IETS methods, namely Unsupervised CRF [Zhao et al. 2008], ONDUX [Cortez et al. 2010] and JUDIE [Cortez et al. 2011]. As many such datasets can be generated for a same attribute, these experiments consider various configurations for mapping datasets to attributes in different ways. The results obtained indicate that the IETS methods can achieve a very good performance if these datasets have a reasonable number of representative values on the domain of the attributes.

In summary, our contributions are: (1) we propose a novel strategy to generate datasets from Wikipedia to support state-of-the-art IETS methods; (2) we show that this strategy is feasible in practice for obtaining many datasets related to real IETS tasks; and (3) we show that the datasets generated using this strategy lead to high-quality extraction results.

This article is organized as follows. Section 2 gives an overview on how current IETS methods exploit the domain knowledge available in existing datasets. Section 3 discusses related work on the use of Wikipedia as a source of domain knowledge. Section 4 presents our strategy for automatically generating datasets from Wikipedia. Section 5 reports the results of experiments we carried out to validate our strategy and to assess its impact on IETS methods. Section 6 concludes the article and gives directions for future work.

## 2. EXPLOITING DOMAIN KNOWLEDGE IN IETS

### 2.1 IETS Problem and Methods

The problem of information extraction by text segmentation (IETS) consists of extracting semi-structured data records by identifying attribute values occurring in continuous text such as bibliographic citations, product descriptions, classified ads, etc. Currently, the most successful IETS methods are based on learning sequential models, such as Hidden Markov Models (HMM) [Borkar et al. 2001] and Conditional Random Fields (CRF) [Lafferty et al. 2001]. In fact, methods based on CRF are the state-of-the-art, outperforming HMM-based methods in experimental evaluations [Sarawagi 2008; Zhao et al. 2008]. In these models, input texts are considered as sequences of tokens or strings (composed by more than one token) to which labels must be assigned, so that these tokens/strings can then be identified as values of attributes.

Sequential models usually require training input texts in which strings representing attribute values are labeled beforehand. From this training data, a model is generated by learning two kinds of features: *state* or *content* features, which are related to the contents of the tokens/strings, and *transition* or *structure* features, which are related to the location of tokens/strings in the sequence. Once generated, the model is applied to label unseen input texts and, therefore, to identify attribute values on these texts.

### 2.2 Learning Content Features from Previous Data

Unfortunately, preparing a significant amount of training input texts may be very expensive or even unfeasible. Thus, to alleviate the need for manually labeled training data, it is possible to use preexisting datasets to learn certain features, as long as their contents are from the same domain as the target textual corpus. This idea has been largely used by several recent IETS methods [Mansuri and Sarawagi 2006; Agichtein and Ganti 2004; Zhao et al. 2008; Cortez et al. 2010; Cortez et al. 2011].

The data stored and the nature of these preexisting datasets depend on each method. For instance, Mansuri and Sarawagi [Mansuri and Sarawagi 2006] proposed a method that uses words stored in dictionaries. Unsupervised CRF [Zhao et al. 2008] requires full records stored in reference tables. ONDUX [Cortez et al. 2010] and JUDIE [Cortez et al. 2011] rely on sets of *attribute values* stored on a *knowledge base*. To simplify the terminology, we will use, from now on, the term *knowledge base*, often abbreviated as *KB*, to refer to all these kinds of preexisting datasets.

Notice that, in all cases, the knowledge bases implicitly encode *domain knowledge*. Thus, they are a very suitable source for learning content features.

### 2.3 Exploiting Domain Knowledge with Content Features

Different aspects of the domain knowledge encoded in knowledge bases are exploited by current IETS methods. Among these aspects we may cite:

*Vocabulary.* Most IETS methods employ features that exploit the common vocabulary shared often by values of textual attributes (e.g., names of neighborhoods, streets, authors, recipe ingredients, etc.). To take advantage of such common vocabulary, there are feature functions based on the similarity between strings in the input and words in dictionaries [Agichtein and Ganti 2004; Mansuri and Sarawagi 2006; Zhao et al. 2008], or on the probability of a string in the input, given the *whole* set of values of an attribute in the KB [Cortez et al. 2010; Cortez et al. 2011].

*Value Range.* For numeric attributes (e.g., page numbers, year, house number, price, etc.), there are features that evaluate how close a numeric string in the input is from the mean value of a set of numeric values of an attribute in the KB according to its probability density function [Cortez et al. 2010; Cortez et al. 2011].

*Format.* The common style often used to represent values of some attributes is also considered. Feature functions based on this aspect evaluate how likely are sequences of symbols forming a string in the input text. For this, typical sequences of symbols occurring on the values of an attribute in the KB are learned. By using such features, it is possible to capture specific formatting properties of URLs, e-mails, telephone numbers, etc. In early methods, these features were computed over training data [Agichtein and Ganti 2004; Mansuri and Sarawagi 2006; Zhao et al. 2008]. More recently, it has been shown that it is also possible to compute them over data on the KB [Cortez et al. 2011; Toda et al. 2010].

A very important point to stress regarding content features is the fact that they can be computed from previously available knowledge bases and, thus, they are independent of the target input text corpus, that is, these features are *input-independent*. In contrast, structure features, which depend mostly on the placement of data within the text inputs, can only be learned from samples from each target corpus. Thus, structure features are *input-dependent*.

### 2.4 Inducing Structure Features from Content Features

Recent methods [Agichtein and Ganti 2004; Zhao et al. 2008; Cortez et al. 2010; Cortez et al. 2011] have shown that input-independent content features learned from a knowledge base can be used to bootstrap the learning of input-dependent structure features. The methods described in [Agichtein and Ganti 2004] and [Zhao et al. 2008] assume that attribute values in the input text follow a single global order. This order is learned from a sample batch of input texts. In [Cortez et al. 2010] and [Cortez et al. 2011], given an input text, an initial extraction is processed using only content features. Assuming that these features are enough to obtain a reasonably correct result, the set of extracted records can be used as training data for learning structure features. In particular, experiments demonstrate that these methods are able to derive accurately probabilities on the sequencing and positioning of strings representing values of the attributes within input texts from a target source. Thus, no fixed-order is assumed.

We notice that, as these methods learn content features from a preexisting knowledge base and use such features to generate training data automatically for learning structure features, no manually labeled sequences are ever required. Thus, these IETS methods are regarded as *unsupervised*.

## 3. WIKIPEDIA AS A SOURCE OF DOMAIN KNOWLEDGE

As discussed above, the existence of a representative knowledge base on the domain of the target input texts corpus is an important requirement for unsupervised IETS methods. A natural question that arises is how such knowledge bases can be generated at a reasonable cost, and, preferably, automatically. A possible answer to this question is using some of the large public knowledge repositories available nowadays on the Web to populate knowledge bases. Wikipedia stands out as one of the most comprehensive and popular among these repositories.

Using the knowledge implicitly available on Wikipedia for Web information retrieval and Web mining tasks is by no means new. In fact, it has been extensively used as a source of instances from which domain knowledge can be automatically acquired in recently proposed methods for solving several different problems such as searching [Arguello et al. 2009], text classification [Wang and Domeniconi 2008; Fuxman et al. 2009; Overell et al. 2009], clustering [Hu et al. 2008; Hu et al. 2009] and semantic enrichment [Weikum and Theobald 2010].

In the information extraction realm, Wikipedia has been used to provide training instances from which features are learned by several Named Entity Recognition (NER) methods recently proposed in the literature [Ratinov and Roth 2009; Weld et al. 2009]. However, to the best of our knowledge, ours is the first proposal for applying such knowledge in IETS methods.

At this point, it is useful to highlight important distinctions between these types of methods. In NER, the problem is to identify, in the input text, strings that refer to real-world entities that belong to few classes, e.g., persons, places and companies. The NER tasks are usually domain-independent (i.e., *open*), because the goal is to identify entities, in general, of certain classes. For instance, these methods seek extracting names of persons in general, such as movie stars or soccer players. IETS methods, on the contrary, aim at extracting values of several different attributes from distinct domains. For instance, in the experiments we report in this article, we carry out IETS tasks involving 13 attributes of three distinct domains. Further, IETS tasks are usually domain-dependent. For instance, different models must be generated for extracting attributes and records of bibliographic citations and product offers.

## 4. GENERATION OF KNOWLEDGE BASES

In our work, we define a knowledge base as a set of pairs $K = \{\langle A_1, O_1\rangle, \ldots, \langle A_n, O_n\rangle\}$ in which each $A_i$ is a distinct attribute, and $O_i$ is a set of strings $\{o_{i,1}, \ldots, o_{i,n}\}$ called *occurrences*. Intuitively, $O_i$ is a set of strings representing plausible or typical values in the domain of attribute $A_i$.

For instance, a very simple example of a knowledge base that includes only two attributes, *Author* and *Title*, is the following:

$$
\begin{aligned}
K = \{ \quad & \langle Author, O_{Author}\rangle, \langle Title, O_{Title}\rangle\} \\
O_{Author} = \{ \quad & \text{"J. K. Rowling","Galadriel Waters", "Beatrix Potter"}\} \\
O_{Title} = \{ \quad & \text{"Harry Potter and the Half-Blood Prince",} \\
& \text{"A Guide to Harry Potter", "The Rabbit's Halloween"}\}
\end{aligned}
$$

Given a data source on a certain domain which includes values associated with fields or attributes, building a knowledge base is a simple process that consists in creating pairs of attributes and sets of occurrences.

In here, we claim that Wikipedia can be used as a source for creating knowledge bases. Specifically, given an XML dump $W$ from Wikipedia, we define a *source* $S$ for an attribute $A$ in a KB $K$ as an

XPath expression over $W$ that generates a set of atomic values in the same domain as $A$. In our work we consider only three types of sources: *Categories*, *Infobox Fields* and *Citation Fields*, which we describe below.

*Categories.* Categories are groups of articles on related topics**. Sources of this type are composed by titles of articles from a given category, e.g., Turing award laureate or Food ingredients;

*Infobox Fields.* Infoboxes are tables that are structured as a set of pairs ⟨field,value⟩. They are available in most articles and present a summary of the information available in the article***. Infoboxes of a same category often follow a *template* that defines which fields are to be used for the category. Sources of this type include the values of a given field in the infoboxes of all articles from this category where this field is available. Examples are Released in infoboxes of articles in the Film category and Pages in infoboxes of articles in the Articles category;

*Citation Fields.* Articles may cite other articles or external sources (books, published articles, etc.). This can be represented by adding a *citation* using a proper template from Wikipedia†. Sources of this type include the values of a given field from this template in all articles containing it. Examples are Journal, Pages, etc.

Simple XPath expressions can be used to generate sources of these types as follows:

—Categories: `/mediawiki/page[category=`$C$`]/title`
—Infobox Field: `/mediawiki/page[category=`$C$`]/infobox/`$F$
—Citation Field: `/mediawiki/page/citation/`$F$

where $C$ is a category name and $F$ is a field name.

Notice that in an actual Wikipedia XML dump, infoboxes and citations are structured using an internal format adopted by the Wikimedia platform, instead of XML. Thus, similarly to what is done in the DBPedia project‡, we use a procedure for pre-processing the XML dump and convert infoboxes and citations to XML. Therefore, in the XPath expressions above, `mediawiki`, `page`, `category` and `title` are actual elements in a Wikipedia XML dump, while elements `infobox` and `citation` are automatically generated by this procedure. This procedure also generates an element $F$ for each field, whose content is the value of the field.

To generate a knowledge base, sources from Wikipedia must be mapped to attributes. In general, several sources can be mapped for a same attribute. They are called *candidate sources* for the attribute. We notice that mapping sources to attributes is actually an instance of the more general problem of schema mapping. In this article, we do not tackle this problem, but the literature is abundant in schema mapping methods that could be adapted for this instance [Halevy et al. 2006]. A deeper investigation of how this can be accomplished is left as future work. In the following, we discuss concrete examples of possible mappings.

Tables I, II and III present, for different attributes of three knowledge bases, examples of candidate sources taken from Wikipedia. These knowledge bases belong to different domains previously used for experimental evaluations of IETS methods: Bibliographical Data, Cooking Recipes and Product Offers. To give a notion of the volume of information available on the sources, these tables also present the number of values and terms (words composing the values) available in each source. Within a same

**http://en.wikipedia.org/wiki/Wikipedia:FAQ
***http://en.wikipedia.org/wiki/Help:Infobox
†http://en.wikipedia.org/wiki/Citation
‡http://dbpedia.org/About

attribute, possible sources are ordered by their number of values. Besides providing real examples, Wikipedia sources presented in this table will also be used in the experimental evaluation we report in Section 5.2.

| Attribute | Source | Values | Terms |
|---|---|---|---|
| **Bibliographic References** | | | |
| Author | *Cat*:Fellows ACM | 318 | 751 |
| | *Cat*:Turing award laureates | 56 | 131 |
| | *Cat*:Computer science writers | 50 | 121 |
| | *Cat*:Von neumann theory winners | 18 | 46 |
| | *Cat*:Dijkstra prize laureates | 13 | 31 |
| BookTitle | *Cat*:Computer science conferences | 117 | 602 |
| | *Cat*:ACM Special Interest Groups | 17 | 34 |
| Journal | *Cit*:Journal | 530057 | 1570292 |
| | *Cat*:English-language journals | 1732 | 6693 |
| | *Cat*:Academic journals | 1665 | 6332 |
| | *Cat*:Academic journal articles | 37 | 173 |
| | *Cat*:Computer science papers | 12 | 68 |
| Date | *Info*:Film/Released | 4119 | 4637 |
| | *Info*:Template/Start date | 1715 | 3367 |
| | *Info*:Books/Publication | 181 | 224 |
| | *Info*:Template/End date | 4 | 7 |
| Pages | *Cit*:Pages | 23904 | 25446 |
| | *Info*:Books/Pages | 862 | 1034 |
| | *Info*:Articles/Pages | 12 | 14 |
| Title | *Cat*:Computer science books | 106 | 446 |
| | *Cat*:Computer books | 37 | 181 |
| Volume | *Cit*:Comic volume | 482633 | 484755 |
| | *Cit*:Volume | 10639 | 10744 |
| | *Cit*:Journal volume | 1095 | 1095 |

Table I. Possible sources for attributes of the Bibliographic References KB.

| Attribute | Source | Values | Terms |
|---|---|---|---|
| **Cooking Recipes** | | | |
| Ingredient | *Cat*:Food ingredients | 110 | 230 |
| | *Cat*:Flour | 90 | 213 |
| | *Cat*:Japanese ingredients | 77 | 121 |
| | *Cat*:Chinese ingredients | 67 | 133 |
| | *Cat*:Indian ingredients | 41 | 66 |
| Quantity | *Info*:Hotel/Number Restaurants | 247 | 261 |
| | *Info*:Religious/Dome Quantity | 110 | 110 |
| Unit | *Cat*:Cooking weights measures | 8 | 16 |

Table II. Possible sources for attributes of the Cooking Recipes KB.

All of the sources fall in one of the three types we consider: *Cat*:$C$ indicates a category $C$, *Info*:$F/C$ indicates a field $F$ from an infobox of the category $C$ and *Cit*:$F$ indicates a field $F$ from citations. $F$ and $C$ were used as parameters in the corresponding XPath expression defined earlier.

The example sources in Tables I, II and III were arbitrarily selected from Wikipedia by considering the domains of the attributes. However, the table shows that, in general, Wikipedia sources provide data in abundance for building KBs. In most cases, categories were used, but numerical attributes are all supplied by infoboxes fields or citation fields.

Notice that Wikipedia receives constant updates and, thus, Wikipedia sources are expected to change over time. While it is desirable to keep KBs up to date with Wikipedia, this is not crucial to IETS methods, since KBs are expected to contain only a sample of the data on the domain of

| Attribute | Source | Values | Terms |
|---|---|---|---|
| **Product Offers** | | | |
| Name | *Cat*:PlayStation 2 games | 681 | 2860 |
| | *Cat*:Wii games | 421 | 1713 |
| | *Cat*:Musical Instruments | 303 | 603 |
| | *Cat*:Ds games | 277 | 1145 |
| | *Cat*:Cookware | 84 | 128 |
| | *Cat*:Appliances | 73 | 174 |
| | *Cat*:Personal computers | 65 | 150 |
| | *Cat*:Portable computers | 33 | 62 |
| | *Cat*:MP3 players | 20 | 37 |
| | *Cat*:Digital camera | 20 | 47 |
| | *Cat*:Consumer electronics | 13 | 29 |
| Brand | *Cat*:Brands | 727 | 1668 |
| | *Cat*:Luxury Brands | 30 | 51 |
| | *Cat*:Electronic companies | 29 | 68 |
| | *Cat*:Brands by Company | 18 | 83 |
| Price | *Info*:Business/Revenues | 49 | 185 |
| | *Info*:Business/Market Value | 15 | 33 |

Table III. Possible sources for attributes of the Product Offers KB.

the attributes. Furthermore, IETS methods do not rely only on KBs, but also on structural features learned from the input texts.

## 5. EXPERIMENTS

In this section, we describe the experiments we have performed to evaluate our strategy using unsupervised IETS methods, namely Unsupervised CRF (U-CRF) [Zhao et al. 2008], ONDUX [Cortez et al. 2010] and JUDIE [Cortez et al. 2011]. First, we describe the experimental setup and the metrics used in the evaluation. Then, we present and discuss the quality of the extraction results obtained. All test datasets, gold standard datasets and knowledge bases are available for download at `http://gtiexperimentos.com.br/wiki-base/`.

### 5.1 Setup

| Dataset | Domain | Text Inputs | Attributes |
|---|---|---|---|
| CORA | Bibliographic References | 500 | 3 to 7 |
| Recipes | Cooking Recipes | 500 | 3 |
| Products | Product Offers | 10000 | 3 |

Table IV. Datasets used in our experiments.

*Datasets.* In Table IV, we present the datasets we used in the experiments. The *CORA* dataset is part of the Cora collection[§] and was used in experiments with several IETS methods [Peng and McCallum 2006; Mansuri and Sarawagi 2006; Cortez et al. 2010; Cortez et al. 2011]. It is composed of a large number of bibliographic citations in distinct styles and formats. The *Recipes* dataset was previously used in [Barbosa and Freire 2010] and [Cortez et al. 2011]. It contains cooking recipes taken from the Web. The *Products* dataset [Cortez et al. 2011] contains product offers from 25 Brazilian e-commerce stores. We notice that, while ONDUX and U-CRF require the input to be provided with individual record explicitly separated, this separation is not required by JUDIE, since it is capable of automatically discovering the structure of the input unstructured records [Cortez et al. 2011].

[§]`http://www.cs.umass.edu/~mccallum/data`

*Reference Knowledge Bases.* To assess the quality of the extraction results obtained with the knowledge bases we generate, we use as reference the results reported in the literature for the IETS methods we tested. For this, we run these methods using the same knowledge used in previous experiments. These reference knowledge bases are presented in Table V.

| Dataset | Source | Used in | Attributes | Records |
|---|---|---|---|---|
| CORA | PersonalBib | [Zhao et al. 2008; Mansuri and Sarawagi 2006; Cortez et al. 2010; Cortez et al. 2011] | 7 | 395 |
| Recipes | FreeBase.com | [Cortez et al. 2010; Cortez et al. 2011] | 3 | 100 |
| Products | Nhemu.com | [Cortez et al. 2010; Cortez et al. 2011] | 3 | 5000 |

Table V. Reference Knowledge Bases.

The data source for building the reference KB used with the CORA dataset is the PersonalBib dataset, which was first used in [Mansuri and Sarawagi 2006]. The reference KB used with the Recipes dataset was built using structured recipes from FreeBase[¶]. In the case of the Products dataset, the reference KB was taken from Nhemu[‖], a Brazilian price comparison website.

*Generated Knowledge Bases.* For each domain, we generated several knowledge bases from the sources presented in Tables I, II and III. For this, we used fixed configurations for mapping candidate sources to the attributes of the knowledge base. These configuration, we call *basic mappings*, are the following: (1) Maximal: each attribute of the KB is associated with the candidate source containing the *highest* number of values; (2) Minimal: each attribute of the KB is associated with the candidate source containing the *lowest* number of values; (3) Full: each attribute of the KB is associated with all candidate sources, meaning that the union of the values of all these sources will be taken as occurrences of the attribute in the KB. The details on the generated KBs are presented in Table VI. For comparison, the reference knowledge bases are also included in this table.

| Knowledge Base | Values | | | Terms | | | Distinct Value Formats | Terms per Value |
|---|---|---|---|---|---|---|---|---|
| | Total | Distinct | Overlap | Total | Distinct | Overlap | | |
| **CORA** | | | | | | | | |
| Reference | 1257 | 1257 | 148 | 7744 | 2360 | 833 | 529 | 2.91 |
| Full | 556001 | 60597 | 237 | 1617258 | 28539 | 1360 | 4846 | 2.94 |
| Maximal | 548683 | 58067 | 232 | 1596098 | 28050 | 1343 | 4733 | 2.91 |
| Minimal | 120 | 116 | 6 | 396 | 241 | 118 | 49 | 2.28 |
| **Recipes** | | | | | | | | |
| Reference | 161 | 161 | 79 | 729 | 191 | 157 | 38 | 1.63 |
| Full | 394 | 378 | 18 | 781 | 511 | 127 | 75 | 1.75 |
| Maximal | 118 | 118 | 8 | 246 | 165 | 70 | 43 | 1.83 |
| Minimal | 49 | 49 | 5 | 82 | 67 | 22 | 10 | 1.28 |
| **Products** | | | | | | | | |
| Reference | 8971 | 8971 | 4311 | 22671 | 9531 | 7576 | 5415 | 4.56 |
| Full | 2796 | 2435 | 451 | 8823 | 3060 | 1045 | 420 | 3.18 |
| Maximal | 1408 | 1408 | 356 | 4528 | 1946 | 781 | 305 | 3.27 |
| Minimal | 43 | 43 | 36 | 80 | 69 | 51 | 51 | 2.91 |

Table VI. Details on the reference and generated knowledge bases.

For each KB, Table VI presents in column "Values/Total" the total number of values taken from the candidate Wikipedia sources and, in column "Values/Distinct", the number of distinct values in these sources, which were actually used to build the KB. In the case of the reference KB, these numbers are the same. The number of distinct values is important to show the diversity in values available in

[¶]http://www.freebase.com
[‖]http://www.nhemu.com

each KB. Column "Values/Overlap" shows the number of values that occur on the KB and that also occur in the corresponding input dataset.

The table also presents information on terms, i.e., tokens that compose the values. Again, the total and distinct number of terms (diversity in terms) taken from the candidate Wikipedia sources (or in the reference KB) are presented in columns "Terms/Total" and "Terms/Distinct" respectively. The term overlap between the KB and the input is shown in Column "Term/Overlap".

The column "Distinct Value Formats" shows the number of distinct value formats found in each KB and the column "Terms per Value" accounts for the average number of terms found in each value. Notice that the information provided in Table VI are indications of the quality of the generated knowledge bases, since these features (terms, values, format) are the main source of information that current unsupervised IETS methods rely on.

Furthermore, this information is strongly related to the aspects of the domain knowledge represented in knowledge bases discussed in Section 2. In the case of textual attributes a high number of distinct terms and a large term overlap with the input dataset favors features related to the vocabulary. In the case of numeric attributes, a high number of distinct values and a large value overlap favors features based on the value range. In both textual and numeric attributes, a high number of distinct values favors features based on the format.

Notice that all numbers in Table VI account for all attributes in each KB. Detailed numbers on each individual attribute are presented in Tables I, II and III

*IETS Methods Implementation.* The implementations of ONDUX and JUDIE were the same used in the experiments reported in [Cortez et al. 2010] and [Cortez et al. 2011], respectively. The U-CRF implementation we use is also the same used in those experiments. It was developed by adapting the publicly available implementation of CRF by Sunita Sarawagi** according to [Zhao et al. 2008] and using additional features described in [Lafferty et al. 2001], e.g., dictionary features, word score functions, transition features, etc. In this case, dictionaries were generated by using values available on the knowledge bases. As required by U-CRF, a batch of input strings is used to infer the order of the attribute values. Based on the configuration used in [Zhao et al. 2008], this batch is built using a sample of 10% of these strings.

*Metrics.* For all performed experiments, we evaluated the extraction results for each individual attribute (attribute-level). As evaluation metrics, we have used the well known precision, recall, and F-measure. Let $B_i$ be a reference set and $S_i$ be a test set to be compared with $B_i$. We define precision $P_i = \frac{|B_i \cap S_i|}{|S_i|}$, recall $R_i = \frac{|B_i \cap S_i|}{|B_i|}$ and F-measure $F_i = \frac{2(R_i.P_i)}{(R_i+P_i)}$. To compute attribute-level results, we calculate precision, recall and F-measure according to the above equations by considering $B_i$ as the set of terms that compose the values of a given attribute $a_i$ and $S_i$ the set of terms assigned to $a_i$ by the IETS method being evaluated.

## 5.2 Results with Basic Mappings

In this section, we present the general quality results obtained in the extraction tasks performed by U-CRF, ONDUX, and JUDIE when using knowledge bases generated using the basic mappings, in comparison to the use of the reference knowledge bases. These results are summarized in Table VII.

As it can be noticed, when using a Full mapping, all methods achieved quality results comparable to the results obtained with the reference KB. The same can be said about the Maximal mapping. Furthermore, it can be seen that the more heterogeneous the KB (Table VI), in the sense of the diversity of terms and values, the better is the extraction quality. While the value overlap between

**`http://crf.sourceforge.net/`

| **CORA** | U-CRF | | | | ONDUX | | | | JUDIE | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Attributes | Ref. | Full | Max. | Min. | Ref. | Full | Max. | Min. | Ref. | Full | Max. | Min. |
| Author | 0.88 | 0.87 | 0.85 | 0.15 | 0.92 | 0.99 | 0.98 | 0.22 | 0.88 | 0.94 | 0.92 | 0.19 |
| BookTitle | 0.56 | 0.59 | 0.62 | 0.24 | 0.89 | 0.79 | 0.77 | 0.36 | 0.79 | 0.71 | 0.75 | 0.25 |
| Journal | 0.49 | 0.73 | 0.73 | 0.39 | 0.90 | 0.91 | 0.90 | 0.66 | 0.86 | 0.87 | 0.89 | 0.54 |
| Date | 0.55 | 0.55 | 0.52 | 0.10 | 0.91 | 0.72 | 0.69 | 0.28 | 0.84 | 0.70 | 0.72 | 0.20 |
| Pages | 0.50 | 0.62 | 0.54 | 0.40 | 0.85 | 0.81 | 0.81 | 0.47 | 0.90 | 0.86 | 0.84 | 0.42 |
| Title | 0.69 | 0.73 | 0.74 | 0.23 | 0.79 | 0.80 | 0.82 | 0.56 | 0.86 | 0.79 | 0.75 | 0.52 |
| Volume | 0.43 | 0.66 | 0.62 | 0.29 | 0.96 | 0.85 | 0.82 | 0.63 | 0.87 | 0.75 | 0.73 | 0.62 |
| **Average** | 0.59 | 0.68 | 0.66 | 0.26 | 0.89 | 0.84 | 0.83 | 0.45 | 0.86 | 0.80 | 0.80 | 0.39 |

| **Recipes** | U-CRF | | | | ONDUX | | | | JUDIE | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Attributes | Ref. | Full | Max. | Min. | Ref. | Full | Max. | Min. | Ref. | Full | Max. | Min. |
| Quantity | 0.91 | 0.82 | 0.80 | 0.73 | 0.97 | 0.86 | 0.85 | 0.79 | 0.96 | 0.90 | 0.89 | 0.75 |
| Unit | 0.93 | 0.91 | 0.89 | 0.85 | 0.95 | 0.93 | 0.93 | 0.90 | 0.94 | 0.90 | 0.91 | 0.92 |
| Ingredient | 0.95 | 0.88 | 0.85 | 0.77 | 0.97 | 0.94 | 0.92 | 0.83 | 0.96 | 0.93 | 0.90 | 0.86 |
| **Average** | 0.93 | 0.87 | 0.85 | 0.78 | 0.96 | 0.91 | 0.90 | 0.84 | 0.95 | 0.91 | 0.90 | 0.84 |

| **Products** | U-CRF | | | | ONDUX | | | | JUDIE | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Attributes | Ref. | Full | Max. | Min. | Ref. | Full | Max. | Min. | Ref. | Full | Max. | Min. |
| Name | 0.84 | 0.85 | 0.83 | 0.70 | 0.91 | 0.91 | 0.90 | 0.76 | 0.90 | 0.91 | 0.88 | 0.74 |
| Brand | 0.80 | 0.73 | 0.71 | 0.70 | 0.89 | 0.91 | 0.92 | 0.82 | 0.92 | 0.87 | 0.83 | 0.83 |
| Price | 0.83 | 0.87 | 0.87 | 0.84 | 0.94 | 0.92 | 0.92 | 0.90 | 0.95 | 0.94 | 0.95 | 0.91 |
| **Average** | 0.82 | 0.81 | 0.80 | 0.74 | 0.91 | 0.91 | 0.91 | 0.83 | 0.92 | 0.91 | 0.89 | 0.83 |

Table VII. Quality of extraction results achieved with the basic mappings.

the input text and the KB does not have high influence over the extraction quality, the term overlap plays a very important role. Thus, the slight advantage of the reference knowledge bases in the majority of the cases is explained by the larger term overlap these knowledge bases present.

Interestingly, there were cases in which the Maximal mapping led to better results than the Full mapping. This is the case of the "BookTitle" attribute in the CORA dataset for all three IETS methods. This can be explained by the fact that some sources may contain incorrect information, which can negatively impact the IETS methods.

As expected, the Minimal mapping led to the worst results, since the knowledge bases built using this mapping configuration presents less data to support the learning of content-related features. This problem has more impact on the CORA dataset, due to the higher number of attributes and to the irregularities in structure of the bibliographic references.

### 5.3 Results with Random Mappings

In the previous experiment, we ran the three IETS methods using the basic mappings, which represent simple fixed heuristics. However, in practice, there could be other approaches for establishing such mappings. For instance, some well-know schema mapping method could be adopted for this task [Halevy et al. 2006].

In such cases, many other mapping configurations, different from the basic ones, might be used. Since it would be unfeasible to anticipate such configurations, in this section we report experiments in which one candidate source is randomly selected for each attribute to compose a knowledge base. We call them *random mappings*. These experiments are intended to evaluate the performance of each IETS method when relying on KBs that could contain noise, thus, we can directly see how robust and resilient those methods are.

For this experiment, five different knowledge bases (R1 to R5) were generated for each dataset using

random mappings. Notice that, in cases of attributes that have a single candidate source (see Tables I, II and III), this source is used in all five knowledge bases generated.

The results are presented in Figure 1 for each dataset by means of the average attribute f-measure achieved.

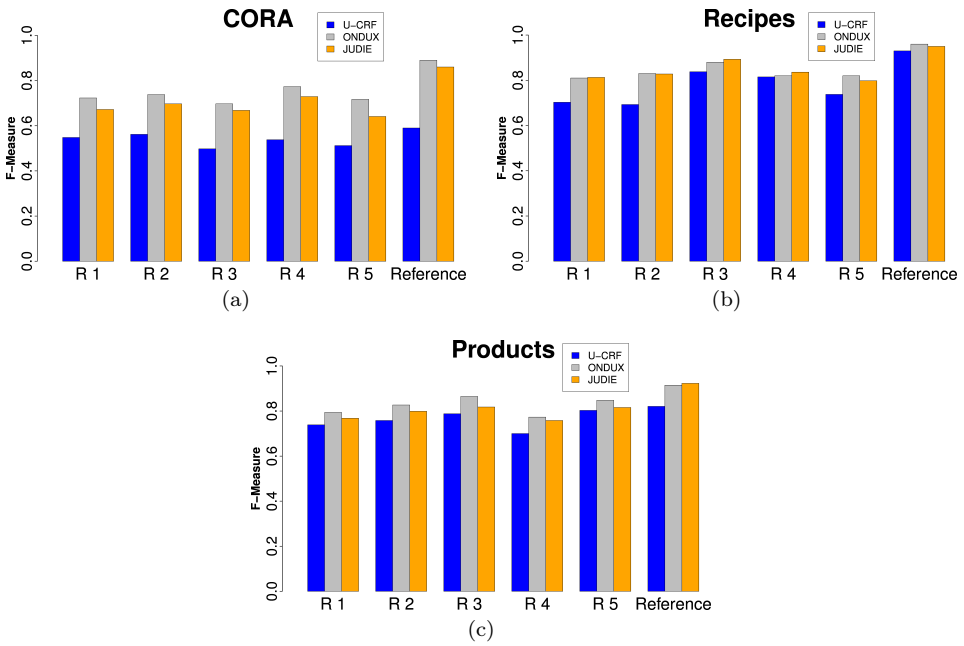


Fig. 1. Quality of extraction results achieved with the random mappings.

As it can be noticed in Figure 1(a), a good extraction quality with ONDUX and JUDIE was obtained with all random mappings in CORA dataset. Extracting from CORA is harder than extracting from the other two datasets. Recipes and Products datasets have only 3 attributes, while CORA records have 3 to 7 attribute and 33 different citation styles [Cortez et al. 2011]. Finally, in Figure 1(b) and (c) all methods were able to achieve high-quality results in all knowledge bases, when compared to the reference knowledge base.

## 6. CONCLUSIONS AND FUTURE WORK

In this article, we proposed a novel strategy that uses Wikipedia for automatically generating knowledge bases to support unsupervised IETS methods. From such knowledge bases, IETS methods can automatically learn content-based features (e.g., vocabulary, value range, format). This not only alleviates the need for manually labeled training data, but also turns the learning process independent from the input texts and yields the unsupervised learning of structure-related features.

Despite their importance, there has been no proper discussion in the literature on how such knowledge bases can be effectively obtained or built in the context of IETS methods. This article takes a first step towards this direction. Experiments we carried out with three stat-of-the-art IETS methods indicate the feasibility and the effectiveness of the proposed strategy.

The work we presented here immediately leads to a number of possible future works. Among them, we may cite: (1) methods for selecting, given an information extraction task, a few candidate sources from a massive number of sources available on Wikipedia; (2) methods for mapping these candidate sources to the attributes of a knowledge base, possibly adapting existing schema mapping approaches

in the literature; and (3) methods for automatically detecting possible noisy information from sources generated with our strategy.

## REFERENCES


Agichtein, E. and Ganti, V. Mining Reference Tables for Automatic Text Segmentation. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Seattle, USA, pp. 20–29, 2004.

Arguello, J., Diaz, F., Callan, J., and Crespo, J.-F. Sources of evidence for vertical selection. In *Proceedings of the International ACM SIGIR Conference on Research & Development of Information Retrieval*. Boston, USA, pp. 315–322, 2009.

Barbosa, L. and Freire, J. Using Latent-structure to Detect Objects on the Web. In *Proceedings of the Workshop on Web and Databases*. Indianapolis, USA, pp. 1–6, 2010.

Borkar, V., Deshmukh, K., and Sarawagi, S. Automatic Segmentation of Text into Structured Records. In *Proceedings of the ACM SIGMOD International Conference on Management of Data Conference*. Santa Barbara, USA, pp. 175–186, 2001.

Cafarella, M., Halevy, A., Wang, D., Wu, E., and Zhang, Y. Webtables: Exploring the power of tables on the web. *Proceedings of the VLDB Endowment* 1 (1): 538–549, 2008.

Cortez, E., da Silva, A., Gonçalves, M., and de Moura, E. ONDUX: On-Demand Unsupervised Learning for Information Extraction. In *Proceedings of the ACM SIGMOD International Conference on Management of Data Conference*. Indianapolis, USA, pp. 807–818, 2010.

Cortez, E., da Silva, A. S., , de Moura, E. S., and Laender, A. H. F. Joint unsupervised structure discovery and information extraction. In *Proceedings of the ACM SIGMOD International Conference on Management of Data Conference*. Athens, Greece, pp. 541–552, 2011.

DeRose, P., Shen, W., Chen, F., Lee, Y., Burdick, D., Doan, A., and Ramakrishnan, R. Dblife: A community information management platform for the database research community. In *Proceedings of the Biennial Conference on Innovative Data Systems Research*. Asilomar, USA, pp. 169–172, 2007.

Fuxman, A., Kannan, A., Goldberg, A. B., Agrawal, R., Tsaparas, P., and Shafer, J. Improving classification accuracy using automatically extracted training data. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Paris, France, pp. 1145–1154, 2009.

Halevy, A., Rajaraman, A., and Ordille, J. Data integration: the teenage years. In *Proceedings of the VLDB International Conference on Very Large Data Bases*. Seoul, Korea, pp. 9–16, 2006.

Hu, J., Fang, L., Cao, Y., Zeng, H.-J., Li, H., Yang, Q., and Chen, Z. Enhancing text clustering by leveraging wikipedia semantics. In *Proceedings of the International ACM SIGIR Conference on Research & Development of Information Retrieval*. Singapore, pp. 179–186, 2008.

Hu, X., Zhang, X., Lu, C., Park, E. K., and Zhou, X. Exploiting wikipedia as external knowledge for document clustering. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Paris, France, pp. 389–396, 2009.

Lafferty, J., McCallum, A., and Pereira, F. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the International Conference on Machine Learning*. Williamstown, USA, pp. 282–289, 2001.

Lange, D., Böhm, C., and Naumann, F. Extracting structured information from wikipedia articles to populate infoboxes. In *Proceedings of the International Conference on Information and Knowledge Management*. Toronto, Canada, pp. 1661–1664, 2010.

Mansuri, I. R. and Sarawagi, S. Integrating Unstructured Data into Relational Databases. In *Proceedings of the IEEE International Conference on Data Engineering*. Atlanta, USA, pp. 29–41, 2006.

Overell, S., Sigurbjörnsson, B., and van Zwol, R. Classifying tags using open content resources. In *Proceedings of the International Conference on Web Search and Web Data Mining*. Barcelona, Spain, pp. 64–73, 2009.

Peng, F. and McCallum, A. Information extraction from research papers using conditional random fields. *Information Processing and Management* 42 (4): 963–979, 2006.

Ratinov, L. and Roth, D. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Conference on Computational Natural Language Learning*. Stroudsburg, USA, pp. 147–155, 2009.

Sarawagi, S. Information extraction. *Foundations and Trends in Databases* 1 (3): 261–377, 2008.

Toda, G., Cortez, E., da Silva, A. S., and de Moura, E. S. A probabilistic approach for automatically filling form-based web interfaces. *Proceedings of the VLDB Endowment* 4 (3): 151–160, 2010.

Wang, P. and Domeniconi, C. Building semantic kernels for text classification using wikipedia. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Las Vegas, USA, pp. 713–721, 2008.

Weikum, G. and Theobald, M. From information to knowledge: harvesting entities and relationships from web sources. Indianapolis, USA, pp. 65–76, 2010.

Weld, D. S., Hoffmann, R., and Wu, F. Using wikipedia to bootstrap open information extraction. *ACM SIGMOD Record* 37 (4): 62–68, 2009.

Zhao, C., Mahmud, J., and Ramakrishnan, I. Exploiting structured reference data for unsupervised text segmentation with conditional random fields. In *Proceedings of the SIAM International Conference on Data Mining*. Atlanta, USA, pp. 420–431, 2008.