Domain-Based Sense Disambiguation in Multilingual Structured Data

Gábor Bella and **Alessio Zamboni** and **Fausto Giunchiglia**[1]

Abstract. Natural language text is pervasive in structured data sets—relational database tables, spreadsheets, XML documents, RDF graphs, etc.—requiring data processing operations to possess some level of natural language understanding capability. This, in turn, involves dealing with aspects of diversity present in structured data such as multilingualism or the coexistence of data from multiple domains. Word sense disambiguation is an essential component of natural language understanding processes. State-of-the-art WSD techniques, however, were developed to operate on single languages and on corpora that are considerably different from structured data sets, such as articles, newswire, web pages, forum posts, or tweets. In this paper we present a WSD method that is designed for short text typically present in structured data, applicable to multiple languages and domains. Our proof-of-concept implementation reaches an all-words F-score between 60% and 80% on both English and Italian data. We consider these as very promising first results given the known difficulty of WSD and the particularity of the corpora targeted with respect to more conventional text.

1 INTRODUCTION

While current formal or semi-formal data models—spreadsheets, XML trees, RDF graphs, etc.—were designed to ease the processing of data by machines, structured data sets still tend to contain a large amount of informal text expressed in natural language within schema elements, data values, and metadata.

Ever more often, applications need to exploit data sets—link, integrate, query, and search them—facing various aspects of diversity in textual data in the process, e.g., the diversity of the languages used, of terminology, or of the domains covered. Picture multilingual Switzerland where a French application on tourism may need to use travel information available in German as well as geographical open data in English, needing to connect data in multiple languages and from multiple domains. Another example is medical data of patients being exchanged across countries for research purposes, again expressed in different languages and using different national standards.Fig. 1 shows examples of natural language text content extracted from real-world data sets that we will use as running examples:

- open government data in Italian and English from the tourism domain containing points of interest in Trento;
- healthcare data in English containing dosages of drugs;
- university data in English and French on papers published by staff containing abstracts, keywords, titles, etc.

Techniques such as cross-lingual semantic matching [2], semantic search [7], or semantic service integration [13] were designed to tackle diversity in data and therefore invariably have some kind of built-in meaning extraction capabilities. In semantic search, natural language queries should be interpreted and matched to data in a robust way so that search is based on meaning and not on surface forms of words (a tourist's query on *'bars'* should also return establishments indicated as *'winebar'*, cf. fig. 1 a, but preferably no results on the physical unit of pressure). In classification tasks, natural-language labels indicating classes need to be formalised and compared to each other (establishments categorised as *'malga'*, i.e., Alpine huts specific to the region of Trento, should be classified as lodging facilities, cf. fig. 1 a). In service integration, on the schema level, attribute names need to be mapped using schema matching techniques (the English *'address'* mapped to the Italian *'indirizzo'*); while on the data level, heterogeneous terminology used across data sets needs to be mapped to common meanings in order to allow interoperability (*'PhD thesis'* equivalent to *'doctoral thesis'*).

We argue that sense disambiguation that relies on conventional natural language processing methods and toolkits, while still applicable, is suboptimal for dealing with diversity in structured data. First, NLP tools and resources tend to be developed for single languages and the representations they use for word senses do not always allow cross-lingual interoperability. Secondly, the type of text appearing in structured data is considerably different from those targeted by state-of-the-art NLP tools. Most existing efforts on NLP concentrate either on 'conventional' text with full, grammatically correct sentences and standard orthography (e.g., newswire, encyclopedia entries, literature, general web content) or on short and noisy text (e.g., tweets, forum comments).

Compared to these cases, text in structured data tends to be shorter and follows different conventions of orthography and syntax. We believe the best-fitting linguistic category to be that of *block language*, defined in [4] as *'abbreviated structures in restricted communicative contexts, especial use being made of the word or phrase, rather than the clause or sentence.'* In such text *'communicative needs strip language of all but the most information-bearing forms'* [3].

The result is that techniques and resources typically used in NLP, such as machine learning models trained on 'conventional' corpora, can only be applied to structured data with a loss in accuracy. Retraining is not in itself a satisfying answer as *sequence labelling* on words—the usual mode of operation of machine learning tools in NLP—relies on an adequate amount of *co-text*, i.e., preceding and following words, the lack of which in structured data again leads to lower accuracy.

[1] University of Trento, via Sommarive 5, 38123 Trento, Italy. {gabor.bella, alessio.zamboni, fausto.giunchiglia}@unitn.it.

Nome	Categoria	Desc_EN	Indirizzo
AI VICOLI	ristorante	Restaurant and Winebar	Piazza Santa Teresa Verzieri, Trento
ORSO GRIGIO	ristorante	typical restaurant	Via degli Orti, 19, Trento
MALGA CANDRIAI	malga	Alpine hut with typical restaurant	Strada di Candriai, 2, Monte Bondone

(a) Open data from Trentino, Italy, on points of interest.

active_substance	name	note	dosage
apixaban	Eliquis	5 mg of 60 film coated tablets	10 milligrams oral
warfarin	Coumadin	30 tablets 5 mg	7.5 milligrams oral, 7.5 milligrams injected

(b) A data set from a healthcare agency on available drugs and their dosages.

Title	Abstract	Date	Type	Keywords
Concept Search: Semantics-Enabled Information Retrieval	The goal of information retrieval is to map a natural language query…	2010	PhD thesis	semantic search - information retrieval - classification
L'oublie la présence le jeu		2009	article	théâtre - comédie - représentation

(c) A data set of university publications.

Figure 1. Simplified examples extracted from real-world data sets, showing various types of natural language text commonly found in structured data.

This paper provides an approach to word sense disambiguation that is adapted to text in structured data and is based on the following principles.

A language-independent representation of meaning. The disambiguation method is designed to be applicable to multiple languages. The hypothesis underlying this design choice is that meanings of words can efficiently be represented as language-independent concepts. We tackle multilingual diversity at design time through the use of multilingual NLP preprocessors, followed by a language-agnostic WSD method that operates on the concept level and can thus be applied to any language.

Domain-Based WSD suits structured data. The backbone of our method is a domain-based WSD algorithm, for two reasons. First, we observe that contents of structured data tend to be domain-specific. Secondly, we argue that formalising the notion of domain is the first step towards tackling this aspect of diversity in structured data. We tackle the diversity of domains at runtime through automated domain extraction and domain-based WSD.

Weaker reliance on co-text. Due to the shortness of text, the output of machine learning methods trained on long text using sequence labelling—such as state-of-the-art part-of-speech taggers—has to be considered as less reliable and 'taken with a pinch of salt.' For this reason we only very minimally rely on co-text in our WSD approach.

Stronger reliance on structural context. Instead of relying on surrounding words, the context encoded in the surrounding data structures (data set, records, attributes) is exploited for additional clues in order to help disambiguation.

The rest of the paper is organised as follows. Section 2 provides a succinct description of linguistic and structural features of text commonly appearing in structured data. Section 3 develops the general architecture and a theoretical description of the solution, as well as implementation details. Section 4 provides evaluations. Section 5 discusses results and problems yet unsolved. Finally, section 6 presents related work.

2 TEXT IN STRUCTURED DATA

2.1 Linguistic Features

In this section we briefly examine the linguistic characteristics of text in structured data.

Languages. It is not uncommon for data sets to mix languages if they were aggregated from heterogeneous sources or if they were produced in geographical areas or usage domains where multilingualism is common practice. The language may change across records (fig. 1 c) or across attributes (fig. 1 a).

Text length. The typical length of textual attribute values is that of a single phrase with less than 10 tokens (words and punctuation). In attribute names 1–3 tokens are typical.

Orthography. The divergence from standard orthography is considerable:

- capitalisation is used arbitrarily: *ALL CAPITALS* or *no capitals* are frequent, as is *Capitalisation Of Each Word* (all of which can be found in fig. 1 a); capitals are therefore not reliable linguistic indicators and, worse, they can confuse machine learning components trained on conventional text,
- punctuation is often omitted or inconsistently used (e.g., dashes instead of commas are used to separate enumerated items, fig. 1 c),
- abbreviations are frequent, especially in attribute names (fig. 1 a),
- in attribute names dash, underscore, or *CamelCasing* are often used for word separation (figs. 1 a and b);

Parts of speech. Nouns are the most frequent, followed by adjectives, prepositions, verbs, and adverbs. Verbs are rare and are mostly limited to present or past participle form (*'coated'* in fig. 1 b). Consequently, the ability to perform lemmatisation (and more generally, morphological analysis) on verbs is not as crucial as on nouns.

Syntax. In rare cases, attribute values may contain text consisting of full grammatical sentences including noun and verb phrases (such as abstracts in fig. 1 c). Most pieces of text, however, are non-sentential and can better be described through the linguistic notion of *minor sentence*, with the following specific characteristics:

- they consist either of a single noun phrase (*'ristorante'*) or noun phrases connected by conjunctions (*'Restaurant & Winebar'*, *'théâtre, comédie, représentation'*);
- the noun phrase can be simple or contain embedded prepositional and noun phrases (*'5 mg of 60 film coated tablets'*);
- ellipsis and other forms of compression are frequently used (*'30 tablets* [of] *5 mg'*).

Semantics. Attribute names frequently express atomic concepts (*'name'*) but sometimes also complex concepts (*'English description'*), mostly using common nouns and adjectives. Proper nouns denoting named entities rarely also appear in attribute names (*'resident of Italy'*). In attribute values, we distinguish between those that encode sets of concepts (typically *class*, *type*, *category* attributes such as *'Categoria'* in fig. 1 a), those that encode sets of named entities (*'name'* in fig. 1 a and b), and the more complex case of descriptive text that may contain both (*'Abstract'* in fig. 1 c).

2.2 Structural Features

While individual pieces of text tend to be short and thus offer a limited opportunity for context-based analysis of meaning, the data structure itself proves to be a alternative source of contextual information. One may draw an analogy between discourse or pragmatics across sentences in conventional long text and high-level meaning that can be extracted across pieces of text within the data structure.

Our analysis on using data structures as context is intended to be as general as possible, applicable invariably to tabular, tree-based, and graph-based structures. However, it is possible to develop more fine-grained context extraction techniques adapted to specific data structures such as trees or graphs. We leave this problem as future work, referring the reader to [12] that provides a deep analysis of context extraction specifically for data schemas.

2.2.1 Structural Context of Data Values

In this section we examine the structural elements of data sets that may serve as context for texts appearing as attribute values.

Other values of the same attribute. Textual values from other records for the same attribute[2] can be used to derive contextual information:

- the larger lexical context of values considered together may help disambiguation (in fig. 1 c the word *'article'* may in itself be ambiguous but is less so in the context of the preceding attribute value *'PhD thesis'*;
- values of an attribute tend to fall under the same domain (in fig. 1 b the domain that can be associated to the attribute *'note'* is *'medicine'*, which makes the meaning of *'film'* less ambiguous);
- repetitions of words and phrases are very frequent in structured data (the word *'milligrams'* in fig. 1 b), a phenomenon that may introduce severe bias in WSD algorithms if not properly accounted for.

Attribute names. WSD does not need to be applied to all types of text: for example, while names or addresses may need to be disambiguated as named entities, the meanings of words composing them are irrelevant in a lot of use cases. While *named entity recognition* techniques can be used to detect such cases, attribute names such as *'name'* or *'address'* may also be indicative of values holding named entities that do not need WSD.

Other values of the same record. Textual values of other attributes in the same record may provide useful context for disambiguation. In tables b and c of fig. 1 the record-level context provides further domain-specific vocabulary.

2.2.2 Structural Context of Schema Elements

Schema elements are not named nor formalised the same way across data formats: OWL has *properties* and *classes*, XML has *attributes* and *elements*, spreadsheets have *column headers*, etc. Extraction methods of structural context vary depending on the goals and on the type of structure (relation, tree, graph). [12] presents a unified approach to modelling various data formats and to context extraction, while here we only provide a high-level summary.

We consider the context of a schema element to consist of other schema elements and of metadata describing the element.

Metadata, in the form of natural language descriptions of the schema element, are frequent in ontologies (e.g., annotation properties) and in open data (e.g., DCAT metadata).

In order to extract context from other schema elements, it is common practice to consider those elements that are directly or transitively related to it, possibly within a given distance.

Trees. In a tree-shaped data structure (a classification, an XML or JSON file) the parent-child relation is used to extract the context of a given node. The context is selected from the set of ancestor and descendant nodes, including the root.

Tables. Tabular data structures can be considered as shallow trees, with the root being the name of the table or relation. The context of an attribute (of a column header) consists of the parent, i.e., the root.

Graphs. In graph-shaped ontological schemas (such as OWL ontologies) relations are named and are often qualified with metaproperties (reflexivity, symmetry, transitivity). The degree of freedom to select the relations that are included in the context of a node is thus much higher than in the previous cases. In this paper we do not consider this use case and direct the reader to [12] for more details.

3 WSD ON STRUCTURED DATA

3.1 General Architecture

Based on our analysis in the previous section, and focusing in particular on the use cases evoked in section 1 (semantic search, classification, query answering, data integration), we identify the following types of meaning extraction tasks relevant for structured data (not pretending to be exhaustive):

concept extraction: this semantic-level operation is commonly solved as the NLP task of *word sense disambiguation*;

[2] The term *record* may unintentionally imply a relational data structure. In case of ontological or object-oriented resources the term *instance* might be more appopriate. We do not intend to restrict the scope of our work to a specific type of data model so we use the term *record* in the most general sense possible.

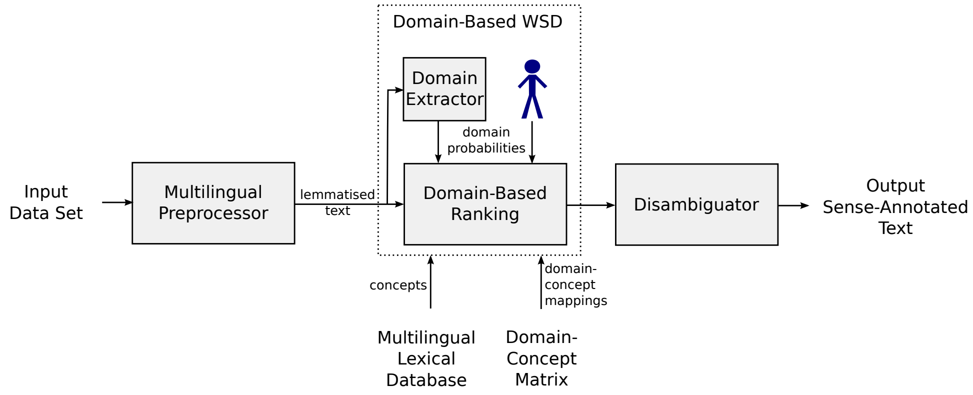


Figure 2. WSD architecture for text in structured data.

named entity extraction: this semantic-level operation is commonly solved as a *named entity recognition and disambiguation* problem;

domain extraction: this pragmatic-level operation is commonly solved as a *document classification* problem and, as we will show, can serve as input for the WSD task.

In this paper we do not consider the extraction of named entities; rather, we set out to provide a WSD method that is able to extract concepts from text appearing in structured data. The high-level architecture of our method is shown in fig. 2:

1. a *Multilingual Preprocessor* provides lemmas and parts of speech for texts in multiple languages extracted from the input data set;
2. possible meanings for lemmas are retrieved from a *Multilingual Lexical Database*;
3. domains relevant to the processed text are estimated by providing *domain probabilities* for lemmatised texts, either by a human user or by a *Domain Extractor* algorithm using a *Domain–Concept Matrix*;
4. based on the domains estimated the *Domain-Based Ranking* of concepts provides preliminary scores to meanings of polysemous words;
5. based on rankings and on hints computed during preprocessing, a *Disambiguator* component produces a final ranking where the top-ranked concepts are the disambiguated ones.

3.2 The Multilingual Lexical Database

WSD annotates words by labels representing formally defined meanings that are usually taken from some form of knowledge resource. In particular, we call *lexical-semantic concepts* the basic meanings defined by two well-known types of linguistically-oriented knowledge resources: *wordnets* and *term bases*. In the case of wordnets the lexical-semantic concept corresponds to the *synset* (i.e., set of synonyms, see [14]) while in term bases it is the *terminological entry*.

The main difference between wordnets and term bases is that the former are single-language multi-domain resources while the latter are (usually) designed to be multi-language and single-domain. Thus, a wordnet maps a lexical entry to one or more possible language-dependent lexical-semantic concepts. These concepts tend to be characteristic of various domains that, however, are usually not explicitly indicated by the wordnet. A term base, on the other hand, maps terms in multiple languages to a single language-independent terminological meaning from a specific domain.

For our purposes of WSD we use a hybrid knowledge resource that we call a *multilingual lexical database* (MLDB). It is defined as a multi-language multi-domain resource where by *multi-language* we understand that it maps lexical entries from multiple languages to language-independent lexical-semantic concepts, and by *multi-domain* we understand that lexical-semantic concepts may belong to different domains. However, we do not require the domains of lexical-semantic concepts to be explicitly indicated within the MLDB.

Let C be an ontology of language-independent lexical-semantic concepts (in short: concepts) c_i. Let l be a lexical entry defined as $l = (\mathfrak{l}, L)$ where $\mathfrak{l}$ is a lemma (word in dictionary form) and L is the language of the lemma. Then the MLDB is defined as

$$\text{MLDB} = \left\{ (l, \{c_i^l\}) \right\}$$

where c_i^l are the language-independent meanings of the lexical entry l.

Existing MLDBs include *EuroWordNet* [17], *MultiWordNet* [15], the *Universal Knowledge Core* (UKC) [5] implemented at the University of Trento and, more recently, *BabelNet* [6]. We used the UKC for our research, also integrating in it some content from MultiWordNet.

In reality, wordnets and MLDBs are more complex than what our definitions above may suggest—in particular, they also encode relations among concepts—but these aspects are not relevant for our paper.

3.3 Multilingual Preprocessing

The MLDB serves the purpose of providing meanings associated to lemmas in multiple languages. Consequently, the text to be sense-disambiguated first needs to be lemmatised. We achieve this using multilingual NLP pipelines specially optimised for the parsing of short text. The languages currently supported are English, Italian, Spanish, and Mongolian. Pipelines consist of the following components, in this order:

Language detector. This component allows the correct language-specific pipeline to be called without an explicit indication of language by the user, which is practical for data sets that contain attribute names or values in multiple languages.

Pipeline selector. Using a heuristic based on text size, one of two pipelines is instantiated:

- a conventional NLP pipeline for longer texts composed of full sentences (we do not discuss this pipeline in the paper), such as abstracts in fig. 1 c;
- a pipeline optimised for short text.

Tokeniser. Tokenisation is optimised to the characteristics of short text as described in section 2.1. We currently use regular expressions but the training of a learning-based tokeniser on short text is also a possibility.

Part-of-speech tagger. As conventional learning-based POS taggers (such as OpenNLP) are suboptimal on short text, we use their output cautiously. First, we distinguish between closed-class and open-class words (nouns, verbs, adjectives, adverbs). For the latter, any further detail provided by the POS tagger is used merely as a hint. Based on prior frequencies of parts of speech we observed in structured data, in cases of polysemous open-class words we use a heuristic scoring system to favour certain parts of speech: nouns $>$ adjectives $>$ verbs. For example, the noun meanings of the word *'search'* in fig. 1 c will be preferred over its verb meanings. Scores are currently hard-coded but in the future we are planning to use syntactic analysis to improve guesses. These scores are used in the final disambiguation phase in combination with domain-based ranking of meanings.

Lemmatiser. Lemmatisation retrieves for every word form all possible lemmas (e.g., *'tablet'* for *'tablets'*. As due to text shortness parts of speech cannot be guessed with a high enough certainty at this point, no POS-based filtering of lemmas is applied.

Multiword detector. Dictionary-based multiword detection is applied to find lemmas composed of multiple words.

Due to the non-conventional syntax of text in structured data (cf. section 2.1), we currently do not apply syntactic parsing. We however plan to research the syntactic properties of such text as future work in order further to improve disambiguation accuracy.

3.4 Domain-Based WSD

The adoption of a domain-based approach as the backbone of our WSD method is motivated by the observation that the contents of structured data sets, and even more of individual attributes within data sets, tend to belong to specific domains. This can be considered as an adaptation of the *one-domain-per-discourse hypothesis* that claims that *'multiple uses of a word in a coherent portion of text tend to share the same domain'* [9, p. 28].

3.4.1 A Formal Notion of Domain

As a simple definition, we represent a *domain label* d_j as a concept taken from the MLDB (such as *'travel'*, *'medicine'*, *'sport'*, *'education'*). We suppose that the set $D = \{d_j\}$ of domains is closed and is relatively small (no more than a couple hundreds), although these are more practical than theoretical requirements.

In conformance to real-world resources, we defined MLDBs not to possess an explicit notion of domain. We therefore provide explicit domain information at this point as an extension to the MLDB. Inspired by [11] and [10] we add domain information to a concept c_i through a mapping to a domain label:

$$m_j = \Big(d_j, \bigcup_{i=1}^{|C|}\{(c_i, w_{ij})\}\Big)$$

meaning that for each concept c_i we provide a weight w_{ij} linking that concept to the domain d_j. The weight w_{ij} is a rational number between 0 and 1. For example, the concept of *'film* [as a movie]*'* will be mapped to the domain label *'media'* with a strong weight while the concept of *'film* [as coating]*'* will be mapped to it with a much lower weight.

The *domain* j is then formally defined as the domain label d_j together with the union of its mappings: $(d_j, \cup_{j=1}^{|D|}\{m_j\})$.

3.4.2 The Domain–Concept Matrix

All mapping of concepts to domains are collected in a resource called the *domain–concept matrix*, $\mathbf{W}$, defined as

$$\mathbf{W} = (w_{ij}) \in \mathbb{Q}^{|C|\times|D|}$$

where $\mathbf{W}$ has as many rows as there are concepts and as many columns as there are domains.

Note that by mapping domain labels to language-independent concepts we obtain a resource $\mathbf{W}$ that can be used across languages.

We are aware of three existing resources mapping meanings from lexical databases to domains:

- *WordNet Domains* (WND) by Magnini et al. [11];
- *Extended WordNet Domains* (XWND) by González-Agirre et al. [10];
- *WordNet Topics* (WNT) included in Princeton WordNet itself starting from version 3.0.

All three use Princeton WordNet as lexical database, thus they can be considered as monolingual English-only resources. WND is an earlier work defining about 170 domains and using binary weights (0 or 1) to model English synsets either belonging or not belonging to domains. XWND was developed as an improved and extended version of WND: it maps *all* concepts to *all* domains using rational numbers for weights, each concept having a positive nonzero weight with respect to each domain. Finally, WNT defines about 440 topics but only annotates a subset of its synsets by topic.

For our work we reused XWND because of its full coverage of WordNet synsets and because of its use of weighted mappings between domains and meanings, lending itself better to statistical methods. For our purposes we modified the XWND resource as follows: first, we converted mappings so that they map domains to language-independent concepts of the UKC instead of English synsets. This way the resource became reusable across languages. Secondly, we made sure that weights of concepts always add up to 1 for any given domain, so that we can consider the set of mapping weights for each domain as a distribution of conditional probabilities $\mathrm{P}(c_i|d_j)$: given a (meaningful) word in a text that we know belongs to domain d_j, $\mathrm{P}(c_i|d_j)$ is the probability of c_i being its meaning. This interpretation allows us to formalise disambiguation using basic probability theory.

3.4.3 Domain-Based Ranking

The domain-based meaning ranking algorithm takes the following inputs:

- the input text as a series of lemmatised tokens;
- the MLDB providing possible concepts for each lemma;
- $\mathbf{W}$ providing domain–concept mappings of the form $\mathrm{P}(c_i|d_j)$;
- a *domain vector* $\bar{d}_t$ that for each domain d_j provides the probability $\mathrm{P}(d_j|t)$ of the input text t belonging to that domain.

Note that having $\bar{d}_t$ as input supposes that the algorithm has prior knowledge of the input domains. This makes sense as data sets are often categorised into domains, e.g., in CKAN open data catalogues, or else they can easily be categorised by the user. Still, in case such information is not available we provide in the next section an automated domain extraction method that computes $\bar{d}_t$ from t.

Based on these inputs, domain-based ranking is provided by the simple formula below that outputs a probability for each concept of each lemma given the input text:

$$\mathrm{P}(c_i^l|t) = \sum_{j=1}^{|D|} \mathrm{P}(c_i^l|d_j)\mathrm{P}(d_j|t)$$

where l is the lemma to be disambiguated and c_i^l are the possible concepts of the lemma provided by the MLDB.

Note that the disambiguation method is context-independent in the sense that it does not take surrounding words into account. This is a deliberate feature that allows the method to work on very short text without affecting performance. Contextual information is present in an implicit manner in the input domain vector $\bar{d}_t$, i.e., in the values $\mathrm{P}(d_j|t)$ that characterise the text as a whole. Note that this supposes that a whole piece of text can entirely be characterised by a single domain vector, in other words, that domains do not change along the text. This is a reasonable hypothesis for short text typically present in structured data.

3.4.4 Domain Extraction

An input domain vector $\bar{d}_t$, providing text-specific domain probabilities to the ranking algorithm, can be obtained in several ways:

1. as a hardcoded default distribution reflecting a prior likelihood of domains to be encountered in data (e.g., an open government data portal is more likely to contain data about tourism or finance than about astrology);
2. as user input, provided either by the data owner (frequent on open data portals) or by the data scientist supervising the meaning extraction task;
3. using an automated domain extraction method.

In this section we provide an algorithm for the third option. Domain extraction can be seen as a document classification problem for which a large number of solutions exist, e.g., supervised learning-based classifiers. Our method is unsupervised and relies only on the same two resources: MLDB and $\mathbf{W}$.

The inputs of the domain extractor are:

- a set of input texts, each as a series of lemmatised tokens;
- the MLDB providing possible concepts for each lemma;
- conditional probabilities $\mathrm{P}(d_j|c_i)$ providing for a concept c_i the probability of it belonging to domain d_j.

Its output is the domain vector $\bar{d}_t$ that represents the probability of each domain being characteristic of text t.

Note that, because single pieces of text are usually too short to provide meaningful domain estimates, the domain extractor is able to take several pieces of text as input simultaneously. In particular, in an analogous manner to the *one-domain-per-discourse* heuristic, we adopt a *one-domain-vector-per-attribute* hypothesis that a single domain vector can be computed over all values of a given structured data attribute combined together. Thus in the following t represents the concatenation of pieces of short text that we suppose to belong to the same domain.

We define the *domain vector of a concept* c as

$$\bar{d}_c = (\mathrm{P}(d_1|c), \ldots, \mathrm{P}(d_j|c), \ldots, \mathrm{P}(d_{|D|}|c)).$$

Intuitively, $\mathrm{P}(d_j|c)$ is the probability of a domain d_j being representative of a concept c.

The domain extraction algorithm estimates the domain vector $\bar{d}_t$ of the input as the centroid of all domain vectors of all possible concepts of all lemmas in text t:

$$\bar{d}_t = \mathrm{centroid}_{\forall c_i^l} \mathrm{P}(d_j|c_i^l).$$

In section 2.2 we observed a problem specific to structured data: repeating words and phrases are very frequent across attribute values. This is a problem as repetitions introduce a significant bias into the computation of centroids and, in general, into any context-based meaning extraction method. Simply removing repetitions, however, would have the adverse effect of giving rare outliers the same importance as to frequently appearing words. As a compromise, we apply a logarithmic function to the number of repetitions, smoothing differences in frequencies all the while favouring frequent words over rare ones.

We still need to show how to obtain $\mathrm{P}(d_j|c_i)$, that is, the domain vector of concept c_i. It is computed from $\mathbf{W}$ using Bayes' theorem:

$$\mathrm{P}(d_j|c_i) = \mathrm{P}(c_i|d_j)\frac{\mathrm{P}(d_j)}{\mathrm{P}(c_i)}.$$

In turn, we need to provide $\mathrm{P}(d_j)$ and $\mathrm{P}(c_i)$.

The former are considered to be *prior domain probabilities* that are initialisation parameters of our WSD method. The user is expected to initialise each $\mathrm{P}(d_j)$ according to their best judgment of domains to appear in input data sets. In the absence of user input, prior domain probabilities can be set to default values, at the worst case as a uniform distribution.

The latter are again computed from $\mathbf{W}$ simply as

$$\mathrm{P}(c_i) = \sum_{j=1}^{|D|} \mathrm{P}(c_i|d_j)\mathrm{P}(d_j).$$

3.5 Disambiguation

Disambiguation is represented as a separate component from domain-based ranking in order to allow a fusion of WSD methods to be applied. In its current version our disambiguator computes final rankings based on two inputs: the output of domain-based ranking and the output of the POS tagger from multilingual preprocessing. Scores output by the former are modified according to the hints provided by the POS tagger, combining the output of an OpenNLP tagger with prior frequencies of POS tags observed in structured data (nouns and adjectives being much more frequent than verbs and adverbs).

Name	Lang.	Content Type	Nb. texts	Nb. tokens
Prodotti Tradizionali Trentini	IT	short and long text from attribute values	108	4,952
Esercizi Alberghieri	IT	short text from attribute values	15	81
Esercizi Alberghieri	IT	attribute names	30	59
Esercizi Alberghieri	IT	short text from metadata values	30	146
Esercizi Alberghieri	EN	short text from metadata values	30	126
Public Infrastructures	EN	attribute names	94	24
Total			**307**	**5,388**

Figure 3. Evaluation data sets taken from open data in Trentino, Italy, and the UK.

4 EVALUATION

4.1 Evaluation Corpora and Method

Our evaluation data sets are open government data from Trentino, Italy[3] and from the UK[4] (fig. 3). The languages tested were English and Italian while the domains covered were food, tourism, and government. Four types of text were analysed:

- long, conventional text as attribute values from the food domain in Italian (*Prodotti Tradizionali*);
- short text as attribute values in Italian from the food domain (*Prodotti Tradizionali*) and from the tourism domain (*Esercizi Alberghieri*);
- attribute names in Italian (*Esercizi Alberghieri*) and in English from the construction domain (*Public Infrastructures*);
- metadata values in Italian from the tourism domain (*Esercizi Alberghieri*) and in English from the construction domain (*Public Infrastructures*).

The corpora were hand-annotated on all words with concepts from the UKC (the multilingual database we used for our evaluations). The English and Italian contents of the UKC were imported from Princeton WordNet 2.1 (110K synsets) and the Italian MultiWordNet (34K synsets), respectively.

Because results were biased by an occasionally incorrect tokenisation, we discarded such tokens from the computation of results. This way we could evaluate the WSD method independently of the rest of the NLP pipeline.

The input domain vectors (i.e., the domains relevant to the data sets) were provided by automated domain extraction, therefore results reflect the performance of the domain extractor and of the disambiguator together, in other words, of the ‘fully automated’ mode of disambiguation without user intervention.

Precision and recall were computed using multi-class evaluation, each concept considered as a class in itself. In order to combine scores of all classes we used both micro- and macro-averaging:

$$P_\mu = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=0}^{|C|} TP_i + FP_i}; R_\mu = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=0}^{|C|} TP_i + FN_i}$$

$$P_M = \frac{1}{|C|}\sum_{i=1}^{|C|} \frac{TP_i}{TP_i + FP_i}; R_M = \frac{1}{|C|}\sum_{i=1}^{|C|} \frac{TP_i}{TP_i + FN_i}$$

Macro-averaging is a simple arithmetic mean over each class, ignoring repeating words (if the same meaning is erroneously disambiguated a hundred times it still counts as one mistake). Micro-averaging, instead, is computed by occurrence and is thus heavily biased by repetitions in the data. Finally, we combined precision and recall for each type of averaging to obtain the F-scores that are shown in the results below.

4.2 Evaluation Results

Our results are shown in fig. 4. There are three pairs of bars shown for each data set, each pair corresponding to the micro- and macro-averaged F-scores:

DB (Domain-Based): our method described above;

Freq (Frequency-Based): as it is common for evaluations of WSD methods, we provide as comparison a baseline frequency-based disambiguator that is based on prior meaning frequencies or ranks, always selecting the most frequent meaning independently of the surrounding text;

KB (Knowledge-Based): still as comparison, a classic knowledge-based WSD method that is designed for longer pieces of text. Using the IS-A hierarchy of concepts in the MLDB, it computes LCA (least common ancestor) distances between the meanings of the word being disambiguated and the meanings of surrounding contextual words. The hypothesis behind this method is that shorter LCA distances correspond to ‘closer’ and thus more probable meanings.

Note that the frequency-based baseline method is context-independent and language-specific, while the knowledge- and the domain-based method share the property of being language-independent as they both operate on concepts. They both rely on context, although in significantly different ways: the knowledge-based disambiguator uses surrounding words (co-text) while the domain-based one uses structural context solely for the purpose of domain extraction.

The following observations can be made about the results:

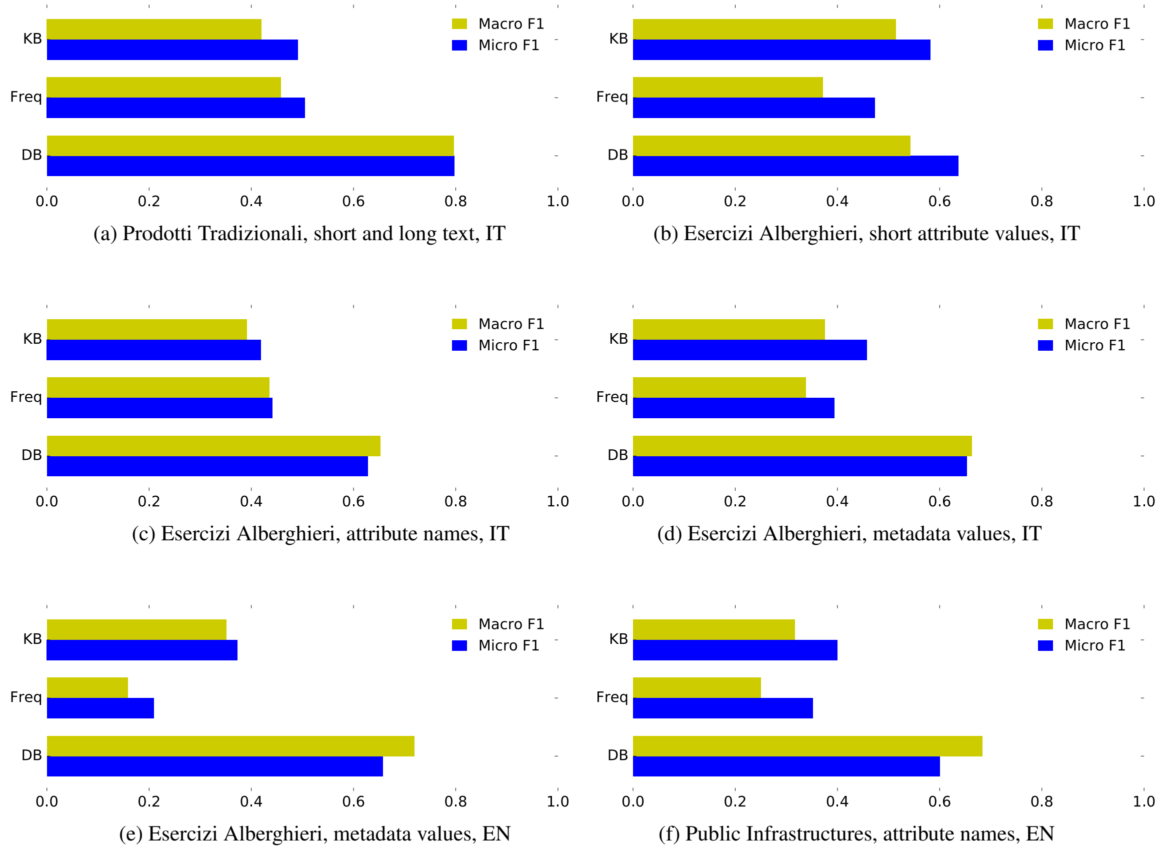
- for all data sets, the scores obtained by our method are superior to the knowledge-based method and the baseline, and except for one data set (b), the difference is consistently higher than 20%;
- there is no significative difference in performance between Italian (a–d) and English (e–f), although results are not directly comparable as the data sets evaluated are different;
- results are the best (80%) on the data set containing large quantities of long text (a): this is not surprising as this data set contains vocabulary that very clearly belongs to the food domain; moreover, larger quantities of text allow the domain extractor to be more precise;
- on short text the micro-F scores are in the 60–65% range while the macro-F scores are more spread out in the 55–72% range.

5 DISCUSSION

Our results seem to confirm that our approach—based on the principles of domain-based operation, a language-independent WSD algorithm, and a structural delineation of context—can suitably support natural language understanding tasks over structured data. The F-scores obtained usually fall in the 60%–70% range (with a lower

[3] `http://dati.trentino.it`

[4] `http://data.gov.uk`



(a) Prodotti Tradizionali, short and long text, IT

(b) Esercizi Alberghieri, short attribute values, IT

(c) Esercizi Alberghieri, attribute names, IT

(d) Esercizi Alberghieri, metadata values, IT

(e) Esercizi Alberghieri, metadata values, EN

(f) Public Infrastructures, attribute names, EN

Figure 4. Evaluation results. KB: knowledge-based disambiguation included for comparison; Freq: frequency-based baseline disambiguation included for comparison; DB: domain-based disambiguation.

outlier of 55% and a higher outlier of 80%) which makes our current WSD implementation usable in real-world applications, including those requiring multilingual and cross-lingual support.

To put our results into perspective, *inter-tagger agreement* and baseline methods provide upper and lower bounds on the scores that can reasonably be targeted by WSD systems. Inter-tagger agreement on all-words tasks (where all meaningful words are sense-annotated) using fine-grained senses (such as those provided by WordNet) was reported in [1, section 1.6] to vary between 70% and 90%. On the other end of the spectrum, a baseline method that always chooses the statistically most frequent meaning reportedly ([1, section 1.6]) reaches 57% on average for the English all-words task on long text. Interestingly, our evaluation of frequency-based baseline disambiguation on our own corpora performed considerably worse, in the 20–40% range. We attribute this somewhat surprising outcome to the domain-specificity of our corpora where the distribution of meanings may be radically different from the corpora used to compute frequency data. The conclusion we draw from this comparison is that the classic baseline method tends to provide scores on structured data that are below the barrier of usability, providing a further argument for more sophisticated WSD methods. Let us also note that, in practice, meaning frequencies are not easily available for languages other than English.

Despite the promising results, we still consider our method to be essentially an early-stage proof of concept. From a research perspective, several of our hypotheses need further verification, as are some rudimentary techniques in need of a more solid theoretical backing.

In particular, a line of research we wish to pursue is a statistically backed-up linguistic analysis of the lexical categories and of the syntax used in block language typically present in structured data. We wish to investigate to what extent this language (or these languages) can be systematically characterised, and to what extent such characterisations may be exploited for WSD, e.g., through statistical parsing. Such a work would be the continuation of successful prior research on *descriptive phrases*, the language of classification labels that has already been described in [7]. We consider descriptive phrases as a special case of block language, and thus a subset of the language we are interested in interpreting.

Another hypothesis that we have not thoroughly investigated so far is the cross-lingual applicability of domain information. The domain resource we exploited—XWND [10], itself derived from WND [11]—was developed using semi-automated knowledge-based techniques on top of Princeton WordNet. It is therefore necessarily biased towards the English language to some extent. While our successful application of it to WSD on Italian does provide a certain evidence towards cross-lingual usability from a practical perspective, it is hard to draw theoretical conclusions from such high-level quantitative comparisons. For instance, a lot depends on the degree of polysemy present in the lexical databases of the languages being compared: a lower number of polysemous words makes the disambiguation task easier. Specifically in the case of English and Italian wordnets the degree of polysemy turns out to be almost identical,[5] so we do not consider our results to be heavily biased by the underlying lexical databases. Still, the linguistic specificity of domain resources remains a question to be investigated, in our view much related to the more general problem of transferring knowledge resources across languages and cultures.

6 RELATED WORK

Word sense disambiguation is a mature research area with a wide range of solutions proposed [1]. While the problem in its generality is considered AI-hard, its actual difficulty is largely dependent on the targeted coverage (lexical-sample vs all-words), granularity of meaning distinctions (homonymy vs polysemy), corpora, etc.

Most research efforts and evaluations, including those reported in [1], were performed on conventional long text. Statistics derived from those results cannot directly be compared to ours, obtained on structured data. Unfortunately, there is very little published research on sense-disambiguation of structured data or block language, making the comparison of our results difficult. Works we are aware of are only concerned with the disambiguation of data schemas, most frequently for the purpose of ontology matching. [8], for example, analyses labels of tree-shaped classifications and proposes a structure-based disambiguation technique taking ancestor and descendant nodes as context. [16] is a survey on similar techniques. In our view, however, ontology matching is not among the tasks that greatly benefit from WSD, as the goal of ontology matching is to find correct matches between ontology elements, regardless of whether their textual contents are correctly disambiguated or not. For this reason, techniques proposed specifically for ontology matching tasks tend not to generalise well to other use cases.

The article [12] provides a detailed analysis on context extraction and disambiguation from data schemas. From our perspective, its main contribution is the generic and adaptable process by which context can be extracted from diverse schema types and depending on the underlying use case. Its difference with respect to our work is that it is aimed at English only, it does not address the disambiguation of data values, and it uses different WSD techniques.

Previous results on domain-driven WSD heavily inspired our work. We adapted some techniques put forth in works by Magnini et al., such as [11], and we reused resources provided by González-Agirre et al. [10]. While these works were developed for the processing of conventional text in English, we wished to show that they could successfully be adapted to structured data and to text in multiple languages.

ACKNOWLEDGMENTS

We acknowledge the *ESSENCE*, *Smart Society*, *Open Data Trentino*, *Healthcare Data Safe Havens*, and *Digital University* projects for having provided us with resources and data. We also acknowledge the authors of the *Extended WordNet Domains* project for having published their resource free of use.

REFERENCES

[1] Eneko Agirre and Philip Edmonds. *Word Sense Disambiguation: Algorithms and Applications*. Springer Publishing Company, Incorporated, 1st edition, 2007.

[2] Gábor Bella, Fausto Giunchiglia, Ahmed Ghassan Tawfik AbuRa‘ed, and Fiona McNeill. A Multilingual Ontology Matcher. In *Proceedings of OM-2015 located at ISWC 2015, CEUR-WS vol. 1545*.

[3] D. Biber, S. Johansson, Geoffrey Leech, S. Conrad, and E. Finegan. *Longman Grammar of Spoken and Written English.* Longman, 1999.

[4] D. Crystal. *Dictionary of Linguistics and Phonetics*. The Language Library. Wiley, 2011.

[5] Fausto Giunchiglia et al. Faceted Lightweight Ontologies. In *Conceptual Modeling: Foundations and Applications*, volume 5600. Springer Berlin Heidelberg, 2009.

[6] Maud Ehrmann et al. Representing Multilingual Data as Linked Data: the Case of BabelNet 2.0. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014), Reykjavik, Iceland, May 26-31, 2014.*

[7] Fausto Giunchiglia, Uladzimir Kharkevich, and Ilya Zaihrayeu. Concept search. In *Proceedings of the 6th European Semantic Web Conference on The Semantic Web: Research and Applications*, ESWC 2009 Heraklion, pages 429–444, Berlin, Heidelberg, 2009. Springer-Verlag.

[8] Fausto Giunchiglia, Mikalai Yatskevich, and Pavel Shvaiko. Semantic Matching: Algorithms and Implementation. *J. Data Semantics*, 9:1–38, 2007.

[9] Alfio Gliozzo and Carlo Strapparava. *Semantic Domains in Computational Linguistics*. Springer-Verlag Berlin Heidelberg, 2009.

[10] Aitor González-Agirre, German Rigau, and Mauro Castillo. *A Graph-Based Method to Improve WordNet Domains*, pages 17–28. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[11] Bernardo Magnini, Carlo Strapparava, Giovanni Pezzulo, and Alfio Gliozzo. Using Domain Information for Word Sense Disambiguation. In *The Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems*, SENSEVAL ’01, pages 111–114, Stroudsburg, PA, USA, 2001. Association for Computational Linguistics.

[12] Federica Mandreoli and Riccardo Martoglia. Knowledge-based sense disambiguation (almost) for all structures. *Inf. Syst.*, 36(2):406–430, April 2011.

[13] Fiona McNeill, Paolo Besana, Juan Pane, and Fausto Giunchiglia. *Service Integration through Structure-Preserving Semantic Matching*, pages 64–82. IGI Global, 2010.

[14] George A. Miller. WordNet: A Lexical Database for English. *Commun. ACM*, 38(11):39–41, November 1995.

[15] Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. MultiWordNet: developing an aligned multilingual database. In *Proceedings of the First International Conference on Global WordNet*, pages 21–25, 2002.

[16] Joe Tekli. An overview on xml semantic disambiguation from unstructured text to semi-structured data: Background, applications, and ongoing challenges. *IEEE Trans. on Knowl. and Data Eng.*, 28(6):1383–1407, June 2016.

[17] Piek Vossen, editor. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[5] We evaluate the *degree of polysemy* by the proportion of monosemous words, the average number of synsets per word, and the average number of words per synset.