Improving Entity Linking using Surface Form Refinement

Eric Charton[1], **Marie-Jean Meurs**[2], **Ludovic Jean-Louis**[1], **Michel Gagnon**[1]
[1]Polytechnique Montreal, [2]Concordia University
Montreal, QC, Canada
{eric.charton, ludovic.jean-louis, michel.gagnon}@polymtl.ca, marie-jean.meurs@concordia.ca

Abstract

In this paper, we present an algorithm for improving named entity resolution and entity linking by using surface form generation and rewriting. Surface forms consist of a word or a group of words that matches lexical units like *Paris* or *New York City*. Used as matching sequences to select candidate entries in a knowledge base, they contribute to the disambiguation of those candidates through similarity measures. In this context, misspelled textual sequences (entities) can be impossible to identify due to the lack of available matching surface forms. To address this problem, we propose an algorithm for surface form refinement based on Wikipedia resources. The approach extends the surface form coverage of our entity linking system, and rewrites or reformulates misspelled mentions (entities) prior to starting the annotation process. The algorithm is evaluated on the corpus associated with the monolingual English entity linking task of NIST KBP 2013. We show that the algorithm improves the entity linking system performance.

Keywords: Entity Linking, Named Entity Resolution, SemLinker

1. Introduction

The Entity Linking (EL) task consists in linking name *mentions* of named entities (NEs) found in a document to their corresponding entities in a reference Knowledge Base (KB). The entities are usually represented by a key number or Uniform Resource Identifier (URI). Dealing with ambiguity is one of the key difficulties of this task, since mentions are often highly polysemous, and potentially related to many different KB entries. Various algorithms and methods have been proposed to address the ambiguity problem. Most of them involve the use of surface forms extracted from Wikipedia.

Surface forms consist of a word or a group of words that match lexical units like *Paris* or *New York City*. They are used as matching sequences to locate corresponding candidate entries in the reference KB, and then to disambiguate those candidates using similarity measures. In this context, misspelled textual sequences can be impossible to identify due to the lack of available matching surface forms. This results in missing or wrong identification of KB entries by EL systems, since accurate candidates are not submitted to the disambiguation process.

This problem is specifically difficult in the context of noisy documents, as for instance forum posts that contain lots of spelling mistakes, nicknames, and puns.

This paper proposes a method to correct named entities mentions of proper names (persons, organizations or geographically-related entities) according to their possible sources of variations and errors. Proper names found in such context have characteristics that makes them different to general text. While there is only one correct spelling for many common words, it is usual to find several spelling variations for proper names. For example, writers, specifically in noisy context like online forums frequently use nicknames or surnames. It is also common to observe evolution of surface forms over time, or influence of people's cultural background or usage. For instance, numerous lexical and spelling variations of geographic names appear when these names are transcribed from an alphabet to another (like Chinese or Cyrillic to Latin alphabet). These issues make matching of proper names often more challenging compared to matching of general text.

To solve this problem, we propose an algorithm for surface form refinement based on Wikipedia resources. The approach extends the surface form coverage of our EL system, and rewrites or reformulates misspelled mentions prior to starting the annotation process. We applied our algorithm on the EL task corpus from the NIST TAC-KBP (Ji et al., 2010) evaluation campaign. This approach improves the EL process accuracy.

This paper is structured as follows. Section 2. reports on previous works related to the EL problem. The Surface form matching problem is investigated in Section 3. We then describe our Surface Form Refinement algorithm in Section 4. In Section 5. we present the SemLinker system that implements the algorithm. Section 6. describes the application of the proposed algorithm on a standard evaluation corpus from the TAC-KBP evaluation campaign[1]. We present the experiments, discuss the obtained results, and conclude.

2. Related Work

Linking textual mentions to KB entities is related to the *Word Sense Disambiguation (WSD)* problem. WSD (Navigli, 2009) consists in determining which sense of an expression is used when it appears in a particular context. Disambiguation resources must be included in annotation systems to handle the WSD problem. For the named entity recognition task, this resource can be generic and generative. For instance, statistical labeling classifiers can be trained on labeled corpora (see for instance works from (Kazama and Torisawa, 2007) and (Béchet and Charton, 2010) using CRF, SVM, or HMM classifiers). Such

[1] `http://www.nist.gov/tac/2013/KBP/data.html`

classifiers are able to infer a named entity class from a limited set of contexts.

This generative approach does not apply to the EL task where each entity to be linked to a semantic description has a specific word context, marker of its exact identity. Dealing with ambiguity is one of the key difficulties in this task, since entities are often highly polysemous, and potentially related to many different KB entries.

Many propositions have been made to solve the named entity disambiguation problem, as for instance the KIM system (Popov et al., 2003). More recently, several systems have been launched as web services dedicated to EL tasks. Most of them are compliant with new emergent semantic web standards like LinkedData network. AIDA (Hoffart et al., 2011b) is a system that relies on the Wikipedia-derived YAGO2 (Hoffart et al., 2011a) knowledge base. It proposes a disambiguation method that combines popularity-based priors, similarity measures, and coherence. Wikimeta (Charton and Gagnon, 2012) is another system relying on DBpedia. It uses a set of bags of words to disambiguate semantic entities according to a cosine similarity algorithm. The common architecture of those systems consists in a first step of mention detection in sentences. This step detects surface forms that may indicate textual sequences linkable to KB entries. Each of these surface forms is used to extract a set of candidates from the KB. Then, in the disambiguation step, the word context surrounding the detected mention in the text is used to measure its similarity with the bag of words provided by the KB for each candidate. Finally, the KB entry that gets the highest score is considered as potentially reflecting the exact identity of the entity to link.

Those systems have been compared with commercial ones like AlchemyAPI, Zemanta or Open Calais in (Gangemi, 2013). The study showed that they perform differently on various essential aspects of EL tasks (mention detection, linking, disambiguation). This suggests a wide range of potential improvements on many aspects of the EL task, specially on mention detection accuracy.

3. The Surface Form Matching Problem

The surface form matching problem is related to the ability of a system to identify an entity, according to its surface form variability.

3.1. Problem Definition

The generic problem of surface form matching is related to the edit distance problem. Numerous studies have been conducted on the edit distance problem (Navarro, 2001). Initially called the *String-to-String Correction Problem* (Wagner and Fischer, 1974), it is a well known field of investigation for natural language processing applications. The task of matching strings has been explored by many research communities. Various studies can be found in literature related to statistics, databases, artificial intelligence, bioinformatics or language. Each community formulates the problem according to its own needs, and numerous techniques have been proposed (Cohen et al., 2003).

In natural language processing context, the problem is usually designated as edit distance task. It can be considered as a way of quantifying how dissimilar two strings are to one another, for example by counting the minimum number of operations required to transform one string into another, according to their lexical and orthographic differences. A classical application of edit distance in natural language processing is the automatic spelling correction used to determine candidate corrections for a misspelled word. Usually, the selection of the correct word involves the use of a dictionary that contains a correct form, and implies the computation of the distance between the word to correct and its canonical form.

Finding and matching proper names, personal names or named entities (including lexical units related to surface forms of more than one word, like *International Business Machine*) is a specific and difficult problem. An increasing number of information extraction and retrieval applications (text and Web mining, search engine query reformulation, data linkage systems) makes this problem crucial to solve. According to (Lait and Randell, 1996) commenting (Bouchard and Pouyez, 1980) the following common person name variations can be inventoried:

- Spelling variations, including interchanged, misplaced, substituted, added or omitted letters. Example of spelling variations are found in *Guo-feng and Kuo-Feng* (from the name Hua Guofeng), additional letters in *Barack* written *Barrac*, and an omission sample could be *Barac*.

- Phonetic variations, when the phonemes of the name are modified or the structure of the name is substantially altered, sometimes using truncation. Phonetic variations in first names can be very large like in the case of shortening of *Barthelemy* in *Bart*.

- Reduction of double names, when surnames are composed of two elements but only one is kept by usage. For example, a double surname such as *Du Pont-De Nemours* may be given in full, as *DuPont*.

- Double first names, not common in the English language, but more frequent in French or Spanish, for example, names such as *Juan-Manuel* may be given in full, or as *Juan* or *Manuel*.

- Alternate first names is another case that occurs when individuals change their names during their life. In this case, an algorithm must be associated with a knowledge base to identify the variations.

All those possible variations also apply to other named entity classes like organizations or geopolitical entities. For example:

- Spelling variations of city names like in *Pekin* sometimes romanized in *Peking*.

- Double names exist frequently for company names like *Microsoft Corporation*, commonly designated by *Microsoft*.

- Reduction is very common for city names like *Sainte Cecile Les Vignes*, commonly shortened to *Sainte Cecile* or *Rio De Janeiro*, reduced to *Rio*.
- Alternate names are very common for city names like *Pekin* and *Beijing* referring to the same city.

To address the surface form matching problem, state-of-the-art techniques are mostly based on pattern matching (Bartolini et al., 2002), sometimes using approximation methods (Hall and Dowling, 1980), phonetic encoding (Zobel and Dart, 1996), or a combination of these approaches. Authors like (Christen, 2006) have concluded that due to variations and errors in names, exact string matching remains problematic. Experimental comparisons of various techniques on standardized corpora indicate that there is still no clear best technique.

3.2. Surface Form Matching for Entity Linking

In EL systems, availability of an exhaustive resource of surface form candidates is of critical interest for detecting mentions. String to string matching methods described in Section 3. do not solve the surface form identification problem. In EL context, the problem is not only to correct or reformulate a mention, but also to associate the mention to appropriate candidate entities to be linked. Those candidates will be used to conduct the disambiguation process and later to establish the link between the mention and the KB. For this reason in EL systems, a resource of valid surface forms according to a giving entity is commonly used. The higher the coverage of this resource, the more candidate entities are detected. Wikipedia-based corpora are generally used as their internal structure allow them to provide valuable correction, misspelling and alternative writing forms according to a given entity to link.

Recent TAC-KBP evaluation campaigns have been engineered to emphasize the surface form matching problem: the evaluation framework of the EL tasks makes increasing uses of noisy and misspelled mentions that have to be linked. In the TAC-KBP 2013 evaluation corpus, we identified three main cases of mentions to annotate for which no surface form exists in Wikipedia-based resources:

1. An abbreviation which refers to a named entity does not exist in any Wikipedia redirection, disambiguation or interwiki page. For example, a named entity is denoted by an abbreviation like *JGL*, which stands for *Joseph Gordon-Levit*[2].
2. An abbreviation which refers to a named entity exists in Wikipedia, but is redirected to another entity. An example is given by the *IPI*[3] surface form, that refers to *Intellectual Property Institute*. An *IPI* disambiguation page exists in Wikipedia, and allows to collect several full names for this abbreviation (see Figure 1) but does not refer to the *Intellectual Property Institute* page.
3. A mention is misspelled or provided under an uncommon or unconventional surface form, and exists in Wikipedia under a slightly different lexical description. This is the case of query *Bagdahd*, that should refer to the *Bagdad* page in Wikipedia[4].

These three cases cannot be handled by state-of-the-art approaches based on Wikipedia derived content only, since the surface forms collected from the encyclopedia do not match the ones expressing named entities in the document. The algorithm we propose empowers EL systems to handle such cases, and thus improves their performance.

4. Surface Form Refinement Algorithm

We propose a Surface Form Refinement algorithm involving two strategies to improve surface form coverage of EL systems. The first strategy consists in automatically adding additional surface forms generated by heuristics to an existing resource of surface forms. The second strategy involves the introduction of a lexical correction step in the surface form detection process.

4.1. Algorithm Description

Let us consider a candidate mention that we want to link to a KB entry. To find a set of candidate entries in the KB according to its surface form, the proposed algorithm runs as follows:

- Step 1: The candidate mention is submitted to the *Improved Surface Form Detection Module*. If matching surface form candidates are returned, the algorithm proceeds to step 3; else to step 2.
- Step 2: If no candidate is provided by the *Improved Surface Form Detection Module*, the candidate mention is submitted to the *Surface Form Correction Module*.
    - If this module returns suggestions of alternative surface forms, step 1 is repeated using those suggestions to collect candidates.
    - Else the algorithm returns no suggestion and exits.
- Step 3: Disambiguation of candidates and selection of entity link.

We describe below the two components of this algorithm, the *Improved Surface Form Detection Module* and the *Surface Form Correction module*.

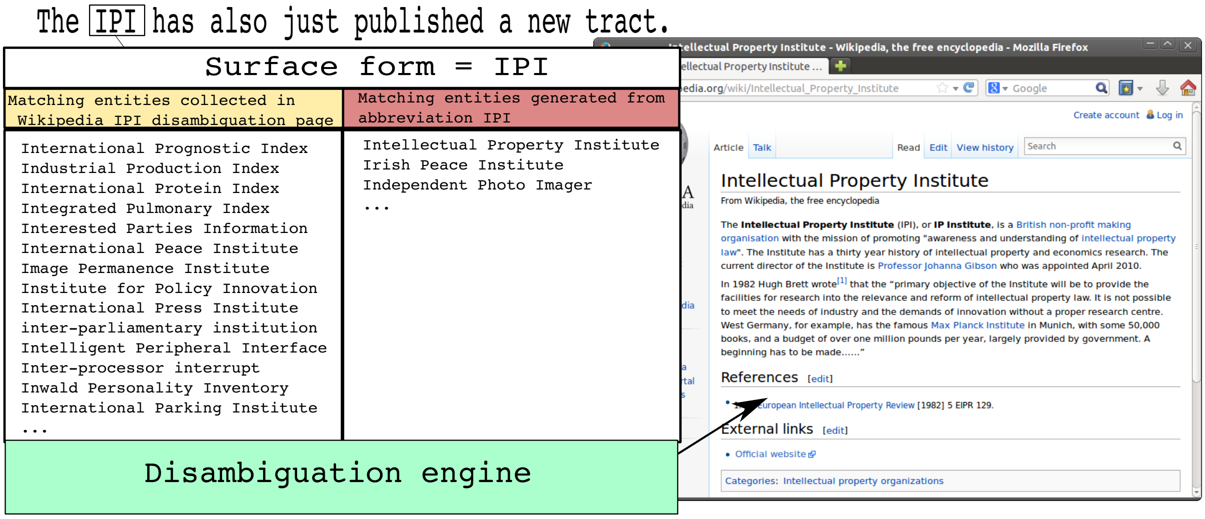
4.2. Improved Surface Form Detection Module

The *Improved Surface Form Detection Module* relies on an enriched set of surface forms. This set is obtained by collecting every Wikipedia internal link that points to an encyclopedic document. This can be *redirection links*, *interwiki links* (directing to the same document in another language edition) and, finally, every *disambiguation page link* that points to encyclopedic documents. To improve the coverage of the resource, the surface forms are collected from

[2] Example from query EL13_ENG_0319 of KBP 2013 corpus

[3] Example from query EL13_ENG_1604 of KBP 2013 corpus

[4] Example from query EL13_ENG_1872 of KBP 2013 corpus


The IPI has also just published a new tract.
Surface form = IPI
Matching entities collected in Wikipedia IPI disambiguation page
Matching entities generated from abbreviation IPI
International Prognostic Index
Industrial Production Index
International Protein Index
Integrated Pulmonary Index
Interested Parties Information
International Peace Institute
Image Permanence Institute
Institute for Policy Innovation
International Press Institute
inter-parliamentary institution
Intelligent Peripheral Interface
Inter-processor interrupt
Inwald Personality Inventory
International Parking Institute
...
Intellectual Property Institute
Irish Peace Institute
Independent Photo Imager
...
Disambiguation engine
Intellectual Property Institute - Wikipedia, the free encyclopedia - Mozilla Firefox
Intellectual Property Institute ...
pedia.org/wiki/Intellectual_Property_Institute
Google
Create account
Log in
Article
Talk
Read
Edit
View history
Search

Intellectual Property Institute

From Wikipedia, the free encyclopedia

The **Intellectual Property Institute** (IPI), or **IP Institute**, is a British non-profit making organisation with the mission of promoting "awareness and understanding of intellectual property law". The Institute has a thirty year history of intellectual property and economics research. The current director of the Institute is Professor Johanna Gibson who was appointed April 2010.

In 1982 Hugh Brett wrote[1] that the "primary objective of the Institute will be to provide the facilities for research into the relevance and reform of intellectual property law. It is not possible to meet the needs of industry and the demands of innovation without a proper research centre. West Germany, for example, has the famous Max Planck Institute in Munich, with some 50,000 books, and a budget of over one million pounds per year, largely provided by government. A beginning has to be made......"

References [edit]

- 1. European Intellectual Property Review [1982] 5 EIPR 129.

External links [edit]

- Official website

Categories: Intellectual property organizations

Figure 1: All the abbreviated surface forms of a given entity are not necessarily present in Wikipedia. Generating some complementary matching (here for the IPI abbreviation) improves the detection capabilities of an Entity Linking System.

six language editions of Wikipedia (English, German, Italian, Spanish, Polish and French). Cross-language resources are extremely useful as in some cases, a surface form only appears in a given language edition of Wikipedia, though it could be used in other languages. Good examples of such cases are the surface forms *Renault-Dacia* or *RNUR* related to an organization that can only be collected from the Polish edition of Wikipedia. Over 10 million surface forms are collected. They allow matching numerous alternate spelling suggestions, including the misspelled ones (usually found in redirection pages of Wikipedia). However, this resource does not cover all the cases encountered in the TAC-KBP corpus.

Hence, additional surface forms are automatically generated with various heuristics like following ones:

- Automatic generation of abbreviations: for example, for a surface form like *Joseph Gordon-Levitt*, we generate the surface form *JGL*.
- Automatic generation of alternative surface forms (like adding "s" at the end of forms)
- Re-ordering n-grams: for example, for a person name like *Barack Obama*, the system generates *Obama Barack*.

We finally obtain 4 millions of additional generated surface forms for a total of 14 millions, each of these related to one or more Wikipedia documents.

According to step 1 of the algorithm described in Section 4.1., for a given word sequence in a text, this resource is utilized to first check if a matching surface form exists. If it exists, all related candidates are collected, and disambiguation is applied to rank them. If no match is provided the *Surface Form Correction Module* is invoked.

4.3. Surface Form Correction Module

The *Surface Form Correction Module* consists of a database of potential spelling errors built from a Wikipedia dump, and a set of rules used to validate the suggested corrections. The database engine generates a set of variations for each existing surface form. The Lucene-Wiki software generates this database[5], which includes about 1 billion entries.

According to step 2 of the algorithm (see Section 4.1.), the *Surface Form Correction module* is called when the *Improved Surface Form Detection Module* does not provide any candidate. If the database engine suggests a rewriting, this refined surface form is submitted to a set of selection rules intended to check if the suggestion is relevant. The rules described below are sequentially applied:

- Rule A : m common word(s). Let us suppose, $m = 1$, and the original surface form is *"hitlery clinton"*[6]: if the system suggests *"Hillary Rodham Clinton"*, the rule selects the suggested refined surface form because it has at least one common word with the original one.
- Rule B : Lexical distance of n letter(s). If $n = 1$, the original surface form is *"Michicgan"*[7], and the system suggests *"Michigan"*, the rule selects the suggested form.
- Rule C: Edit-distance between mention and suggestion. The rule verifies if the original and corrected mentions start with the same character and calculate their l Levenshtein distance. It accepts the suggested form if $l < t$ where t is a threshold value.

If the suggested refined form is accepted, it is submitted to the *Improved Surface Form Resource* to obtain a list of candidates. The disambiguation algorithm is then used to get the first ranked candidate, and then to select a link.

5. Algorithm implementation

The Surface Form Refinement algorithm was integrated in the SemLinker system, presented in the TAC-KBP 2013 evaluation campaign.

[5] https://www.mediawiki.org/wiki/Extension:Lucene-search
[6] Example from query EL13_ENG_0895 of KBP 2013 corpus
[7] Exemple from query EL13_ENG_1624 of KBP 2013 corpus

5.1. SemLinker System

SemLinker is a Java library designed to experiment various applications of text mining and annotations. Using an external annotator (currently www.wikimeta.com), SemLinker can apply various annotation layers and information extraction processes to textual documents. The SemLinker system is detailed in (Charton et al., 2013). The SemLinker system handles a query according to the TAC-KBP format, processes it through a pipeline architecture as follows (see Figure 2 for an architecture overview):

1. Reformulate the query if necessary using the Surface Form Refinement algorithm.
2. Annotate each entity in the document with a ranked set of Wikipedia URIs (the candidates) using an external annotator compliant with Wikipedia URIs. NE labels and Part Of Speech (POS) tags are also provided.
3. Re-rank candidates for each entity in the document using all the annotation layers.
4. Extract the best entity link in the document using the query definition and query expansion.
5. Cluster the detected entities with no corresponding KB entries (NILs), and convert the Wikipedia URIs format to TAC-KBP node identifiers.

Four main modules are dedicated to these tasks:

1. *Query Reformulation module* (using Surface Form Refinement algorithm).
2. *Mutual Disambiguation module*.
3. *Link Extraction module*.
4. *Clustering module*.

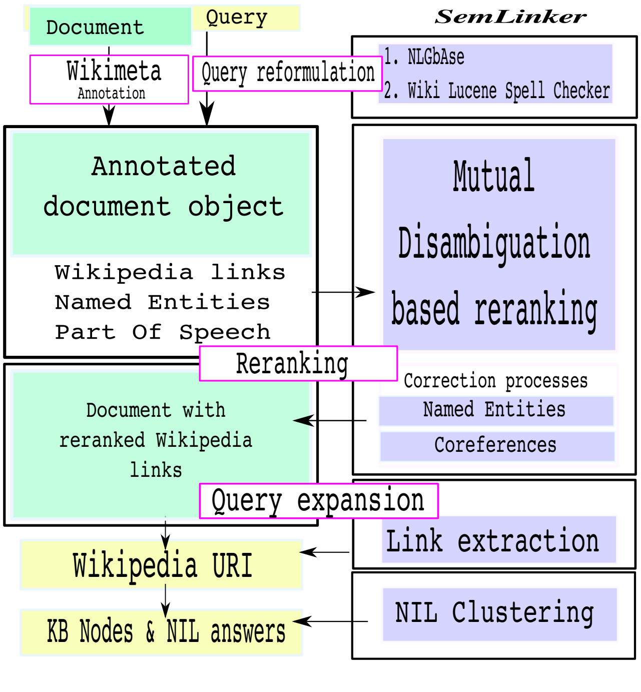


Figure 2: SemLinker architecture.

5.2. NLGbAse lexical resource

The original surface form resource used in this study is NLGbAse, a set of metadata derived from Wikipedia (Charton and Torres-Moreno, 2010). NLGbAse is a multilingual linguistic resource composed of metadata, and built from the Wikipedia encyclopedic content. The structure of Wikipedia, and the sequential process for building metadata (like these recorded in NLGbAse), have been described in (Bunescu and Pasca, 2006). The process is applied in (Charton and Torres-Moreno, 2010).

Currently, this resource contains about 3 millions of English metadata, each of them describing a unique concept. For each concept, a metadata unit contains a bag of words with pre-calculated TF.IDF and possible surface forms of the concept (ie *New-York, NYC, Big Apple*).

For each document in Wikipedia, NLGbAse provides a set of metadata, composed of three elements: (i) a set of surface forms, (ii) all the words contained in the document, where a TF.IDF weight (Salton and Buckley, 1988) is assigned to each word, (iii) a NE class tag (person, org...) obtained through a classification process.

The set of surface forms is obtained through the collection of every Wikipedia internal link that points to an encyclopedic document. For instance, the surface form set for the NE *Paris (France)*[8] contains 39 elements, (eg. *Ville Lumière, Ville de Paris, Paname, Capitale de la France, Département de Paris*). NLGbAse is involved in SemLinker as a resource for the Surface Form Refinement algorithm in the *Query Reformulation module*.

6. Experiments and Results

The proposed algorithm has been evaluated on the TAC-KBP EL task. The basic principle of KBP task is to provide a query for a given document. This query contains a surface form related to an entity to annotate, a reference of the document and the exact position of the surface form in the document. For a given query, the task is to determine the correct node in the reference KB for the entity, and to add a new node for the entity if it is not already in the reference knowledge base (KB). The reference KB is derived from an October 2008 dump of English Wikipedia, which includes 818,741 nodes. Entities can be of type person (PER), organization (ORG), or geopolitical entity (GPE). The document collection built for previous KBP evaluation campaigns contains a combination of newswire articles (News), posts to blogs and newsgroups (Web). In 2013, discussion fora were added to the previous categories. Table 1 provides a breakdown of the queries per categories of entities, and per type of documents.

Category	All	PER	ORG	GPE	News	Web	Forum
# queries	2190	686	701	803	1134	343	713

Table 1: Breakdown of the TAC-KBP 2013 test corpus queries according to entity types, and document categories.

[8] http://www.nlgbase.org/perl/display.pl?query=Paris&search=FR

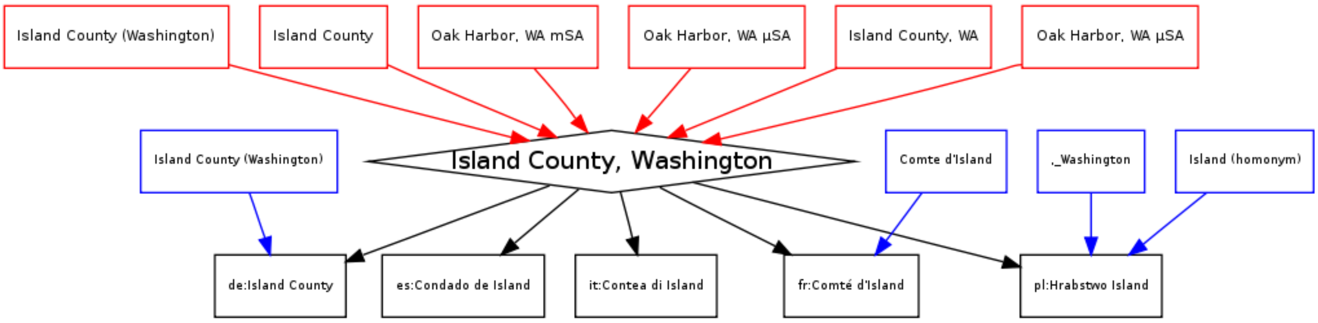


Figure 3: In this example, surface forms for a county name are collected from several language editions of Wikipedia.

	refSF	**QR**
Category	$B^3 + F_1$	$B^3 + F_1$
Overall	0.574	**0.596**
KB (in KB)	0.494	**0.535**
NIL (not in KB)	0.665	0.662
NW (news doc)	0.645	0.649
WEB (web doc)	0.579	0.592
DF (forum doc)	0.454	**0.508**
PER (person)	0.695	0.708
ORG (organization)	0.604	0.607
GPE (geopolitical entity)	0.440	0.486

Table 2: System performance on test corpus of KBP2013 task with reference surface form resource (refSF) and improved surface form detection algorithm (QR). The B^3+F1 metric used is the official one from KBP evaluation framework, and is derived from B^3 metric used to evaluate coreference resolution systems.

The algorithm was tested with the TAC-KBP 2013 English queries[9] according to the evaluation protocol of the TAC-KBP task evaluation framework. The B^3 metric evaluates the EL process using the clusters of queries provided by the system for each KB link. We submitted the queries to a version of our EL system with the surface form correction algorithm disabled (refSF system in Table 2), and enabled (QR system in Table 2). With QR system, when a refined surface form is proposed, each of its occurrences in the document is rewritten according to the new form, prior to be submitted to the semantic annotation engine. Our system obtained the results reported in Table 2.

The Surface Form Refinement algorithm improves the performance for KB link detection (KB line of Table 2), and does not significantly reduce the performance for NIL KB (surface form with no matching KB link). We can conclude that the selection rules of the *Surface Form Correction Module* accurately rejects most of the wrong corrections of surface forms. Improvement of performance obtained on the noisiest documents (DF docs of the KBP task are web forum transcripts) also shows that the Surface Form Refinement algorithm is efficient with noisy text content.

7. Conclusion and Future Work

In this paper, we introduced a Surface Form Refinement algorithm to improve EL systems. In the context of a standard evaluation based on NIST TAC-KBP resources, the system using the proposed Surface Form Refinement method outperforms the EL system only based on Wikipedia derived resource for surface form matching.

We have discussed the characteristics of named entity surface form variations in the TAC-KBP evaluation corpus and the potential sources of variations and errors in them. We presented a system for addressing the problem of surface form variations with both pattern matching and distance measure techniques, based on an extensive use of lexical resources generated from Wikipedia encyclopedic content. Experimental results on this data set have shown that the combination of those techniques and resources can improve the performance of a standard EL system. However, named entity matching in the context of the EL task remains a very challenging research field. This work is focused on very specific proper names. While our approach obtains promising results in this context, how it would perform on more specialized entities like common names or specific named entities like biological ones is still to evaluate. An interesting future work could be to study the performance of the proposed algorithm on other topic oriented tasks, like for example biological annotation tasks.

Reproducibility

The system presented in this paper has been submitted to the NIST-TAC KBP 2013 evaluation campaign under the system reference *polymtl*. The presented results are the official ones from this campaign. All the experiments presented in this paper are fully reproducible on NIST KBP data using the SemLinker software, publicly released at `https://code.google.com/p/semlinker/`.

Acknowledgments

This research was supported as part of Dr Eric Charton's Mitacs Elevate Grant, sponsored by 3CE. Participation of Dr Marie-Jean Meurs in this work was supported by the Genozymes Project, a project funded by Genome Canada and Genome Quebec. Authors would like to thank Wikimeta Technologies Inc. that provided support and computing resources for this research.

[9] http://www.nist.gov/tac/2013/KBP/data.html

8. References

Bartolini, I., Ciaccia, P., and Patella, M. (2002). String matching with metric trees using an approximate distance. In *String Processing and Information Retrieval*, pages 271–283. Springer.

Béchet, F. and Charton, E. (2010). Unsupervised knowledge acquisition for extracting named entities from speech. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 5338–5341. IEEE.

Bouchard, G. and Pouyez, C. (1980). Name variations and computerised record linkage. *Historical Methods*.

Bunescu, R. C. and Pasca, M. (2006). Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. ACL.

Charton, E. and Gagnon, M. (2012). A disambiguation resource extracted from Wikipedia for semantic annotation. In *Proceedings of LREC 2012*.

Charton, E. and Torres-Moreno, J. (2010). NLGbAse: a free linguistic resource for Natural Language Processing systems. In *Proceedings of LREC 2010*.

Charton, E., Meurs, M.-J., Jean-Louis, L., and Gagnon, M. (2013). SemLinker system for KBP2013: A disambiguation algorithm based on mutual relations of semantic annotations inside a document. In *Text Analysis Conference KBP*. U.S. National Institute of Standards and Technology (NIST).

Christen, P. (2006). A comparison of personal name matching: Techniques and practical issues. In *Sixth IEEE International Conference on Data Mining (ICDM) Workshops*, pages 290–294. IEEE.

Cohen, W., Ravikumar, P., and Fienberg, S. (2003). A comparison of string metrics for matching names and records. In *KDD Workshop on Data Cleaning and Object Consolidation*, volume 3, pages 73–78.

Gangemi, A. (2013). A Comparison of Knowledge Extraction Tools for the Semantic Web. In *The 10th Extended Semantic Web Conference (ESWC) 2013*.

Hall, P. A. and Dowling, G. R. (1980). Approximate string matching. *ACM computing surveys (CSUR)*, 12(4):381–402.

Hoffart, J., Suchanek, F. M., Berberich, K., Lewis-Kelham, E., De Melo, G., and Weikum, G. (2011a). Yago2: exploring and querying world knowledge in time, space, context, and many languages. In *Proceedings of the 20th international conference companion on World wide web*, pages 229–232. ACM.

Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., and Weikum, G. (2011b). Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 782–792. Association for Computational Linguistics.

Ji, H., Grishman, R., Dang, H., and Griffitt, K. (2010). Overview of the TAC 2010 knowledge base population track. *Proceedings of TAC 2010*.

Kazama, J. and Torisawa, K. (2007). Exploiting Wikipedia as external knowledge for named entity recognition. In *Proceedings of the 2007 EMNLP-CoNLL*, pages 698–707.

Lait, A. and Randell, B. (1996). An assessment of name matching algorithms. *Technical Report Series-University of Newcastle Upon Tyne Computing Science*.

Navarro, G. (2001). A guided tour to approximate string matching. *ACM computing surveys (CSUR)*, 33(1):31–88, March.

Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10.

Popov, B., Kiryakov, A., Kirilov, A., Manov, D., Ognyanoff, D., and Goranov, M. (2003). KIM – Semantic annotation platform. *Lecture Notes in Computer Science*, pages 834–849.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information processing & management*.

Wagner, R. A. and Fischer, M. J. (1974). The string-to-string correction problem. *Journal of the ACM (JACM)*, 21(1):168–173.

Zobel, J. and Dart, P. (1996). Phonetic String Matching: Lessons from Information Retrieval. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '96, pages 166–172, New York, NY, USA. ACM.