*

MSC^+: Language Pattern Learning for Word Sense Induction and Disambiguation

Fábio Bif Goularte[a,], Danielly Sorato[a], Silvia Modesto Nassar[a], Renato Fileto[a], Horacio Saggion[b]

[a]*Dept of Informatics and Statistics, Federal University of Santa Catarina, Florianópolis, SC, Brazil.*
[b]*Natural Language Processing Group, Department of Communication and Information Technologies, Pompeu Fabra University, Barcelona, Spain.*

Abstract

Identifying the correct meaning of words in context or discovering new word senses is particularly useful for several tasks such as question answering, information extraction, information retrieval, and text summarization. However, specially in the context of user-generated contents and on-line communication (e.g. Twitter), new meanings are continuously crafted as the result of existing words being used in novel contexts. Consequently, lexical semantics inventories and systems have difficulties to cope with semantic drifting problems. In this work, we propose an approach to induce and disambiguate word senses of some target words in collections of short texts, such as tweets, through the use of fuzzy lexico-semantic patterns that we define as sequences of Morpho-semantic Components (MSC). We learn these patterns, that we call MSC^+ patterns, from text data automatically. Experimental results show that instances of some MSC^+ patterns arise in a number of tweets, but sometimes using different words to convey the sense of the respective MSC in some tweets where pattern instances appear. The exploitation of MSC^+ patterns when they induce semantics on target words enable effective word sense disambiguation mechanisms leading to improvements in the state of the art.

Keywords: Lexical semantics, Information extraction, Linguistic pattern mining, Word sense induction, Word sense disambiguation.

1. Introduction

The semantic annotation of words in social media is a challenge. Social media text imposes additional difficulties for automatic methods to carry out quality disambiguation (Camacho-Collados et al., 2016), such as context information quite limited (e.g. short text), poor grammatical rules conformity (e.g. noise), and high redundancy. In addition, a language is a polysemic symbolic system without ready semantics for some constructs.

A word can be interpreted in multiple ways depending on the context in which it occurs – lexical ambiguity phenomenon (Albano et al., 2014; Alagić et al., 2018). Sometimes words have implicit semantics (e.g., to make humor, irony, or wordplay). For example, the tweet *"Blond, brunette or red-headed Devassa ???"*[1] plays with words in English that usually refer to hair color and the word **devassa** from the Portuguese language. The intended meaning of this word in this

*Corresponding author
Email addresses: fabio.goularte@gmail.com (Fábio Bif Goularte), danielly.sorato@posgrad.ufsc.br (Danielly Sorato), silvianassar@ufsc.br (Silvia Modesto Nassar), r.fileto@ufsc.br (Renato Fileto), horacio.saggion@upf.edu (Horacio Saggion)

[1]Capitalizing the first letters of words representing proper names, as in this tweet, is not guaranteed or even common in social media. Thus, one can not rely on this for disambiguation.

context is not the one you may find in a dictionary, but *beer*! It can be inferred by considering the pattern appearing in the tweets presented in Table 2, as explained in the following. We have found that implicit semantics induced by the language patterns investigated in this paper is quite common in colloquial language, and particularly in social media. In cases like this, current annotation methods frequently fail to capture the correct meaning of certain words, leading to results with low levels of precision and recall (Bontcheva and Derczynski, 2016).

Some of the most prominent tasks for modeling and resolving the lexical ambiguity problem are Word Sense Disambiguation (WSD) and Word Sense Induction (WSI) (Navigli and Vannella, 2013; Alagić et al., 2018). Both tasks are fundamental in Natural Language Processing (NLP). WSI automatically discovers the possible senses for target words (Schütze, 1998) in text documents, regarding the context in which each word appears. WSD, in turn, automatically disambiguates the possible meanings to assign the most probable one to each target word. Some methods for WSD rely on a fixed inventories of word senses such as WordNet (Navigli, 2012). Many solutions for other widely used semantic annotation tasks such as Named Entity Recognition/Normalization/Disambiguation (NER/NEN/NED) or Entity Linking (EL) (Zhang et al., 2010; Liu et al., 2012) also rely on pre-defined sense inventories.

However, building and constantly updating large inventories of word senses (e.g., WordNet (Fellbaum, 1998), BabelNet (Navigli and Ponzetto, 2010), Yago (Mahdisoltani et al., 2013)) is an expensive and difficult task. As a result, important current senses of words used in social media, by specific groups, in certain geographic regions or in particular domains may not appear in sense inventories. For example, consider the tweets[2] (i) to (iii) transcribed below and the sense of the word **polar** in each one of them.

(i) And the journey continues: the Arctic Sunrise reaches the **polar** region.

(ii) This is the first-ever view of the **polar** region of Jupiter.

(iii) I wish to drink an ice-cold **polar**.

Table 1: Some senses for the target word "polar"

#	PoS tag	Gloss
1	Adj	Characterized by opposite extremes.
2	Adj	Of or existing at or near a geographical pole or within the Arctic or Antarctic Circles
3	Adj	Extremely cold

Consider the sense inventory extract for the word **polar** presented in Table 1. The intended sense for **polar** in tweets (i) and (ii) is that of line #2 in Table 1, because of clear references to regions of Earth and Jupiter, respectively. However, in tweet (iii), **polar** plays the role of a proper name and the verb **to drink** induces a sense of beverage which is out of the scope of some sense inventories. Difficulties to catch the exact word sense will happen using any sense inventory that does not know about the beer brand called **Polar**. Sometimes, the intended word sense exists in the inventory, but it is hard to devise algorithms that disambiguate to the correct sense. For example, in BabelNet, we found 19 senses (12 nouns and 6 adjectives) for the word **polar**. Two of them related to *beer*. Then, we submitted message (iii) to the Babelfy tool[3], and the word **polar** was annotated with the sense of *arctic* (extremely cold).

In this paper, we propose a new approach and automated methods to induce and disambiguate the correct sense of words in text. As many WSI/WSD approaches, it also takes advantage of contextual information. However, for the best of our knowledge, our proposal is the first one to automatically learn context information from some collection of short text as language patterns defined by sequences of Morpho-Semantic Components (MSC), which we call MSC^+ patterns.

[2]Tweets collected using `https://twitter.com/search-home` on October 2 2018.

[3]Babelfy is a unified, multilingual, graph-based tool that combines Entity Linking and Word Sense Disambiguation based on BabelNet `http://babelfy.org`

Table 2: Example of MSC^+ pattern instances of size 3 (MSC_1, MSC_2, MSC_3) identified on tweets # 1 to 5

#		MSC_1 ⟨Verb, Ingest⟩		MSC_2 ⟨Adj, Any⟩	MSC_3 ⟨Noun, Beer⟩	
1	Hey	**get**	your	**cold**	**beer**	and relax.
2	We could	**drink**	an	**ice cold**	**Budweiser**	or two.
3	I wish to	**drink**	an	**ice-cold**	**polar**	.
4	Let's	**drink**	one more	**cold**	**Devassa**	ok?
5		**Drinking**	a	**blond**	**devassa**	.

Each MSC has the most probable senses (e.g. candidate concepts or named entities) and the most probable morphosyntactic classes (e.g. Part-of-Speech (PoS) tags) for a word in a text document. An MSC^+ pattern can be seen as a sequence of pairs $\langle morphosyntactic\ class, Sense\rangle$ that repeatedly appear in the MSCs of distinct short texts (e.g. tweets). An instance of an MSC^+ pattern is a sequence of not necessarily adjacent but consecutive MSCs (i.e. words) in some text, in which the most probable morphosyntactic class and sense for each MSC match those in the respective position the pattern.

Firstly, we provide formal definitions for MSC and MSC^+ patterns, and an unsupervised algorithm to mine these patterns. Then, we show that these patterns are frequent in tweets. Finally, we exploit these linguistic patterns for doing correct WSI/WSD in cases for which current tools fail, leading to better precision and recall of semantic annotations.

Table 2 provides an example of MSC^+ pattern arising in short texts (#1 to #5) extracted from Twitter[4]. Each one of these tweets (one per line) has an instance of the MSC^+ pattern that is a sequence of distinct MSCs (i.e. words) with respective candidate morphosyntactic classes and senses matching the sequence: $MSC_1\langle Verb, Ingest\rangle$, $MSC_2\langle Adj, Any\rangle$, $MSC_3\langle Noun, Beer\rangle$. Each instance of this pattern is a sequence of 3 MSCs (columns MSC_1, MSC_2 and MSC_3) highlighted in bold in the respective tweet (line). These sequences are similar in terms of the PoS tags and meanings of their respective MSCs. MSC_1 is a verb referring to *liquid ingestion* (sense = Ingest). MSC_2 is an adjective that can refer to *cold*, *extremely cold* or *hair color*, among other possibilities (sense = Any). MSC_3 is a noun, but only for the tweets #1 and #2 its sense is correctly disambiguated to *beer* and *beer brand*, respectively, by using current automatic approaches.

Target words (whose sense has to be solved yet) in the other tweets are highlighted in red. Thanks to the MSC^+ pattern established by the MSC sequences of the first two tweets, our method can also disambiguate them. Notice the partial adherence of the MSC sequence in #3 to this pattern. Each MSC in #3 matches with the respective one in the pattern, except the word **polar** (MSC_3), whose PoS class is also a noun, but whose sense is initially undetermined. This partial adherence to the MSC^+ pattern induces the sense of the MSC_3 from the previous tweets to the word **polar** in tweet #3, and allows it to be disambiguated to *beer*.

Analogously, one can also induce sense to and disambiguate the word **Devassa** to *beer* in tweets #4 and #5, as well as other short texts (from authors with similar language habits) in which such a word appears with the same MSC^+ pattern or a similar one (e.g. the tweet *"Blond, brunette or red-headed Devassa ???"*). We employ this rationale to automatically mine MSC^+ patterns and use them for doing WSI and WSD when there is partial matching with some mined pattern involving a word that current automatic approaches have difficulties to disambiguate.

Some approaches for WSI/WSD learn and employ lexical patterns (Liu et al., 2010). Nevertheless, MSC^+ patterns are an alternative for some situations in which current approaches fail, as exemplified above. These patterns can be used to improve semantic annotations produced by us-

[4]Tweets collected by using https://twitter.com/search-home on November 21 2018.

ing a variety of annotation methods (Moro et al., 2014; Fileto et al., 2015). Then, the improved semantic annotations can help to boost a variety of computational tasks and applications, ranging from text simplification (Saggion et al., 2015), summarizers (Goularte et al., 2019) and knowledge base enrichment (Fellbaum, 1998; Camacho-Collados et al., 2016; Navigli and Ponzetto, 2010; A. Júnior et al., 2015; Ruiz-Casado et al., 2007) to events detection (Xia et al., 2015), sentiment analysis (Dragoni, 2018), and question answering (Al-Harbi et al., 2017).

The main contributions of this paper can be stated as follow:

1. the introduction and formal definition of MSC, MSC sequences, MSC^+ patterns and MSC^+ pattern instances;
2. an approach for word sense induction and disambiguation based on MSC^+ patterns;
3. an algorithm to find the most frequent MSC^+ patterns in a set of documents that have been previously annotated with candidate morphosyntactic classes and candidate senses of semantically relevant words.
4. two methods for word sense induction and disambiguation based on MSC^+ patterns.

The results of experiments reveal major characteristics of MSC^+ patterns mined in a set of tweets, and the contribution of word sense induction and disambiguation based on the mined patterns to improve precision and recall of semantic annotations produced by state-of-the-art systems. They also show that the variation of our method for word sense induction and disambiguation relying on both PoS tagging and word sense confidence leads to the best results.

The remainder of this paper is organized as follows. Section 2 provides some foundations necessary for understanding our proposal. Section 3 discusses related works. Section 4 formally defines key concepts of our approach such as MSC and MSC^+ patterns. Section 5 describes our approach in a top-down fashion, i.e., first in terms of main stages and then details of key tasks. Section 6 and Section 7 report the experiments for performance evaluation and discuss their results. Finally, Section 8 presents conclusions and indications of future work.

2. Preliminaries

This section provides an overview of methods for capturing word senses, the problem tackled in this paper. It also reviews sequence pattern mining, and how it could be used to find textual patterns that are less elaborated than MSC^+ patterns. However, it can help to understand our proposal and it distinguishable traits.

2.1. Capturing word senses with WSI and WSD

A word sense is a discrete representation of one aspect of the meaning of a word (Jurafsky and Martin, 2018). Word Sense Induction (WSI) and Word Sense Disambiguation (WSD) are traditional approaches to automatically capture word senses. WSI, sometimes also called unsupervised WSD, is a NLP task which goal is to classify and identify multiple senses of polysemous words. Most WSD method do not use any predefined sense inventory (Navigli, 2009). WSD automatically assigns one meaning per word in accordance with the context where the words occur. It usually takes such senses from a predefined sense inventory (Navigli, 2009).

Currently, the most used fine-grained sense inventories in studies about lexical semantics[5] are WordNet[6] (Fellbaum, 1998) and BabelNet[7] (Navigli and Ponzetto, 2010). Both provide a wide-coverage semantic network of the word meanings. Despite this fact, these sense inventories do not cover all words and meanings used by humans (e.g., in specific domains such as law or medicine, the use of creative slang for emerging topics at a recent moment, words that assume a new semantic value as a consequence of the context).

In contrast to knowledge-based and supervised approaches, WSI can discover senses of wide cov-

[5]International Workshop on Semantic Evaluation - SemEval, https://aclweb.org/aclwiki/SemEval_Portal

[6]http://wordnet.princeton.edu

[7]http://babelnet.org/stats

erage and high accuracy without manually annotated training data. Instead, a set of "senses" of each word are captured automatically from the instances of each word in the training set (Jurafsky and Martin, 2018). For this reason, WSI is an attractive alternative to WSD for lexical semantics studies. WSI relies on the unsupervised approach and does not use human-defined word senses (Jurafsky and Martin, 2018).

Most algorithms for WSI are typically derived from clustering techniques. For instance, the algorithm of Schutze (Schütze, 1998) whose goal is to represent each word as a context vector of bag-of-words features $\vec{c}$. However, their evaluation is generally more difficult (Navigli, 2012). According to (Jurafsky and Martin, 2018), algorithms for WSI apply three steps.

1. Compute a context vector $\vec{w}$ for each token/word w.
2. Use a clustering technique to build a partition of the word/token context vectors $\vec{w}$ into a predefined number of groups or clusters. Each cluster refers to a word sense.
3. Compute the centroid of each vector cluster. Each vector centroid is a sense vector representing that sense of w.

2.2. Sequential pattern mining

Sequential pattern mining is a data mining task that tries to discover patterns in the form of recurrent subsequences in a set of sequences (e.g. of numbers or symbols) by using statistically measured criteria, frequency of occurrence, length, and profit (Fournier-Viger et al., 2017).

The idea of sequential pattern mining can be transposed to text analysis. In a text, the ordering of words or relevant elements in sentences is important (Bashar et al., 2017). Thus, sentences in a text are considered sequential data, with each one being a subsequence of words (Fournier-Viger et al., 2017). The most frequent word subsequences define patterns that can be used for WSD (Béchet et al., 2012).

For example, Table 3 shows the word patterns found by using the Apriori algorithm[8] in the tweets presented in Section 1. Table 3 shows only the patterns that occur at least two times (minimum support) in (i), (ii), and (iii) tweets. Notice that these patterns only indicate the most recurrent words and the words that appear together most frequently. The support (or absolute support) of a sequence s_a in a document D is defined as the number of sequences that contain s_a, and is denoted by $sup(s_a)$. In other words, $sup(s_a) = |\{s|s \subseteq s_a \wedge s \in D\}|$ (Bashar et al., 2017).

Table 3: Word patterns

Id	Pattern	Support	Message
1	$\langle\{the\}\rangle$	5	i, ii
2	$\langle\{polar\}\rangle$	3	i, ii, iii
3	$\langle\{of\}\rangle$	2	ii
4	$\langle\{region\}\rangle$	2	i, ii
5	$\langle\{polar, region\}\rangle$	2	i, ii
6	$\langle\{the, polar\}\rangle$	2	i, ii
7	$\langle\{the, polar, region\}\rangle$	2	i, ii

The MSC^+ patterns proposed in this work are more elaborate in two senses: (i) they take into account the order of occurrence of the words, not just their co-occurrence in the documents; and (ii) MSC^+ pattern instances are compared in more subtle and abstract ways than word matching. MSC (instead of just words) are compared in terms of PoS tag and sense class.

3. Related Work

The main works are based on four clustering approaches: (i) context, (ii) word, (iii) graphs, and (iv) probabilistic (Navigli, 2012). The standard approach is first one, in which the contexts of word instances are represented as vector space model (e.g., bag-of-words) of first or second-order. The context vectors are obtained by techniques, such as word-context co-occurrence statistics (Schütze, 1998), word embeddings (Li and Jurafsky, 2015). Then, context vectors are

[8]Designed for finding groups of items frequently appearing together - frequent item sets problem.

grouped into sense clusters (e.g., K-means algorithm, Brown algorithm). The second approach consists of clustering words which are semantically similar. Usually, the similarity/relatedness between the words is measured in terms of syntactic dependencies (Lin, 1998). The third approach is based on graph clustering (Panchenko et al., 2017; Rodriguez et al., 2016; Pelevina et al., 2017; Chang et al., 2018; Gutiérrez et al., 2017). According to Di Marco and Navigli (2013), this approach represents word instances similar or related to each target word as nodes in a similarity graph which is grouped using graph clustering algorithms (e.g., HyperLex (Véronis, 2004), PageRank (Agirre et al., 2006), Chinese Whispers (Biemann, 2006)). The fourth approach represents senses by statistical models, such as Bayesian framework (Brody and Lapata, 2009), Bayesian in Learning Word Embeddings (Wu et al., 2018), Hierarchical Dirichlet Process (Lau et al., 2012). First, for each ambiguous word is created a set of senses by the probability distribution over words. Thus, context words are generated according to this distribution, and different senses can be obtained which have different word distributions.

The main evaluation techniques for WSD systems are unsupervised (Alagić et al., 2018; Jurgens and Klapaftis, 2013; Pelevina et al., 2017) and supervised (Jurgens and Klapaftis, 2013; Panchenko et al., 2017; Pelevina et al., 2017). For the unsupervised evaluation, the induced senses (clusters) are directly compared with a gold-standard annotation using measures as paired F-score (Artiles et al., 2009), V-measure (Rosenberg and Hirschberg, 2007), and Fuzzy NMI (Jurgens and Klapaftis, 2013). The paired F-score computes the harmonic mean of precision and recall by treating as true positives all instance pairs that are clustered together in both induced and gold sense clusters (Alagić et al., 2018). V-Measure computes the homogeneity and completeness of clusters. Homogeneity represents if all of its clusters contain only data points which are members of a single class, whereas completeness represents if all the data points that are members of a given class are elements of the same cluster (Rosenberg and Hirschberg, 2007). The second type of evaluation, supervised, the induced senses are mapped to gold standard senses using a mapping heuristic, and precision and recall measures are used to determine the quality of the resulting WSD system. Usually, baseline systems are used to compare the results of WSD systems, such as the Most Frequent Sense (MFS) for each word from a corpus (Jurgens and Klapaftis, 2013; Moro and Navigli, 2015; Alagić et al., 2018). In the WordNet, the MFS is corresponding to the first sense of a word.

The currently best-performing approaches to WSI rely on lexical substitutes (Alagić et al., 2018; Jurgens and Klapaftis, 2013). In this approach, a system creates a substitute vector for each target word from the most likely substitutes suggested by a statistical language model (Navigli, 2012). In SemEval-2013 Task 13 (Jurgens and Klapaftis, 2013), the best performance of WSI systems in the single-sense setting is *F-score* = 0.64. In (Alagić et al., 2018), the system performance using SemEval-2010 dataset is paired *F-score* = 0.58. In SemEval-2015 Task 13 (Moro and Navigli, 2015), the best system for English is able to obtain a performance similar to the systems mentioned above.

The other line of research relevant for our work is WSI based on detecting patterns in lexical data. Lexical pattern approaches for WSD explore grammatical dependencies (e.g., syntactic, morphological, semantic) from words and contexts, or word pairs and patterns. For instance, using syntactic dependency structures automatically derived from text to building a collection of paths sharing distributionally similar nominal anchors (Lin and Pantel, 2001); learning an optimal combination of the various knowledge sources for individual target words (Mihalcea, 2002); abstraction of pattern using features (e.g., lemma and Part-of-Speach) and kernel methods (Strapparava et al., 2004); identify lexical patterns between concepts in online encyclopedia to extend existing ontologies or semantic networks (Ruiz-Casado et al., 2007); using ensembles of Naive bayesian classifiers to extract infrequent sense instance with the same n-gram (Liu et al., 2010). All these works implement a supervised evaluation by classical precision, recall and F1 measures

using data from SemEval (except in (Lin and Pantel, 2001)).

The availability of SemEval datasets boosted works in WSI/WSD based on lexical patterns (Mihalcea, 2002; Strapparava et al., 2004; Liu et al., 2010). For example, the SemEval-2013 Task 13[9] provided three datasets with weighted word sense annotations and labeled with WordNet 3.0 senses. Despite the coverage of SemEval datasets in different domain and languages, most datasets to WSD task are texts from documents, and apart from a few exceptions to WSD task (e.g., multiword expressions and noun and verb supersenses) that consists in social media data[10] (Schneider et al., 2016).

Our approach combines several of the above ideas and adds features ensuring interpretability. Most notably, we use a word sense inventory (Alagić et al., 2018) and tagger/WSD tool to enrichment data from social media with semantic information; for inducing sense we rely on morphosyntactic and semantic context features (Strapparava et al., 2004; Panchenko et al., 2017), co-occurrences (Panchenko et al., 2017) and learning of patterns (Mihalcea, 2002; Ruiz-Casado et al., 2007; Liu et al., 2010).

4. Basic Definitions

4.1. Morpho-semantic components

Some words in text can have their meaning induced and disambiguated by exploiting morpho-semantic patterns. A morpho-semantic pattern is characterized by a sequence of Morpho-semantic Components (MSC) with length based on n-gram. An MSC refers to a word in a document text d and its associated sets of candidate morphosyntactic classes (e.g. from some PoS tag set) and senses (e.g. from some sense class set), as stated in Definition 1.

Definition 1. ***Morpho-semantic Component - MSC****. Given a text document d with identifier idd, a set of morphosyntactic classes Λ and a set of senses Ξ, an MSC is a sextuple $msc = \langle idd, idc, b, e, P, S\rangle$, where:*

idd : is the identifier of the document (e.g., id of the tweet) where msc occurs;

idc : is the identifier of the morpho-semantic component msc;

b : is the beginning of msc in the text document identified by idd;

e : is the end of msc in the text;

$P = \{\langle p_1, \tilde{p}_1\rangle, \ldots, \langle p_n, \tilde{p}_n\rangle\}$ $(n \geq 1)$: is a list of n candidate PoS classes for msc, in which each pair $\langle p_i, \tilde{p}_i\rangle$ indicates candidate class $p_i \in \Lambda$ and its normalized degree of belief $\tilde{p}_i \in (0, 1]$, in such a way that $\sum_{i=1}^{n} \tilde{p}_i = 1$ $\forall \langle p_i, \tilde{p}_i\rangle, \langle p_j, \tilde{p}_j\rangle \in P : p_i \neq p_j$;

$S = \{\langle s_1, \tilde{s}_1\rangle, \ldots, \langle s_m, \tilde{s}_m\rangle\}$ $(m \geq 1)$: is a list of m canditate senses for msc, in which each pair $\langle s_i, \tilde{s}_j\rangle$ indicates a candidate sense $s_j \in \Xi$ with its normalized degree of belief $\tilde{s}_j \in (0, 1]$, in such a way that $\sum_{j=1}^{m} \tilde{s}_j = 1$ $\forall \langle s_i, \tilde{s}_i\rangle, \langle s_j, \tilde{s}_j\rangle \in S : s_i \neq s_j$.

Each candidate morphosyntactic class $p_i \in \Lambda$ and each candidate sense s_j is associated with the respective degree of belief ($\tilde{p}_i$ and $\tilde{s}_j$, respectively), because sometimes the PoS class and the sense of the MSC returned by the annotation tool is uncertain. The set of possible PoS classes and and the set of possible word senses vary according to the annotation tool and the sense inventory employed for text annotation. For instance, ontology classes, resources of linked data, and synsets of lexicons are commonly used to represent word senses. When the value of an annotation of a word is an instance, the word sense can be induced or compared with other senses by considering its class (e.g., beer brand, food, person, institution).

Of course, there are correlations between candidate morphosyntactic classes and candidate senses of words, i.e., certain sense only apply to

[9] `https://www.cs.york.ac.uk/semeval-2013/task13/`, texts extracted from source types such as fiction, journal, letters, non-fiction, technical, travel guides.

[10] `http://dimsum16.github.io/`

some words when the play particular morphosyntactic roles in sentences. However, we prefer to dissociate candidate senses and morphosyntactic classes in MSCs for two reasons: (i) there are tools that tags words just with possible senses or just with possible morphosyntactic classes; (ii) the dynamics of languages can make it dificulty to keep track of all the senses and morphosyntactic classes that can be associated with certain words and the correlations between the respective classes.

The relevance of an MSC in a text document d can determined by applying some summarization method to d, for example, among other possibilities. An MSCs can also be classified in non-disambiguated, weakly disambiguated or strongly disambiguated, depending on the confidence of its candidate senses and morphosyntactic classes. Definition 2 and 3 state the conditions for an MSC to be considered weakly disambiguated and strongly disambiguated, respectively.

Definition 2. ***Weakly disambiguated MSC*** *A morpho-semantic component msc is weakly disambiguated with respect to the parameters τ^p_{abs}, $\tau^s_{abs} \in (0,1]$ if, and only if, there are at least one pair $\langle p, \tilde{p}\rangle \in P$ and at least one pair $\langle s, \tilde{s}\rangle \in S$ such that:*

$$\tilde{p} \geq \tau^p_{abs} \quad \wedge \quad \tilde{s} \geq \tau^s_{abs}$$

Definition 3. ***Strongly disambiguated MSC*** *A morpho-semantic component msc is strongly disambiguated with respect to the parameters τ^p_{abs}, τ^s_{abs}, τ^p_{dif}, $\tau^s_{dif} \in (0,1]$ if, and only if, it is weakly disambiguated with respect to τ^p_{abs} and τ^s_{abs}, and there are one pair $\langle p, \tilde{p}\rangle \in P$ and one pair $\langle s, \tilde{s}\rangle \in S$ such that:*

$$\forall \langle p_i, \tilde{p}_i\rangle \in P | p_i \neq p : \tilde{p} - \tilde{p}_i \geq \tau^p_{dif} \quad \wedge$$
$$\forall \langle s_j, \tilde{s}_j\rangle \in S | s_j \neq s : \tilde{s} - \tilde{s}_j \geq \tau^s_{dif}$$

Notice that a strongly disambiguated MSC has just one candidate sense whose confidence surpass the ones of all the other candidate senses by at least τ^s_{dif} and one morphosytactic class whose confidence surpass the ones of all the other morphosytactic classes by at least τ^p_{dif}. The representation of a strongly disambiguated MSC can be simplified to include only the disambiguated morphosytactic class p and the disambiguated sense s, as follows: $msc\_disambiguated = \langle idd, idc, b, e, p, s\rangle$.

4.2. MSC^+ patterns

Many relevant MSCs can appear in a text document d possibly separated from each other by words considered irrelevant. Definition 4 formally specifies a sequence of MSCs in a text document.

Definition 4. MSC^+ ***Sequence***. *A sequence of morpho-semantic components (MSC^+ sequence) is an ordered list $msc^+ = (msc_1, \ldots, msc_l)$ ($l \geq 1$) of morpho-semantic components referring to subsequent but not necessarily adjacent words in the same document d, i.e., $\forall msc_k = \langle idd, idc_k, b_k, e_k, P_k, S_k\rangle, msc_{k+1} = \langle idd, idc_{k+1}, b_{k+1}, e_{k+1}, P_{k+1}, S_{k+1}\rangle \in msc^+ : b_{k+1} > e_k \;\; (k = 1, \ldots, l)$.*

An MSC^+ sequence is said strongly disambiguated if all its $l \geq 1$ morpho-semantic components are strongly disambiguated according to a set of parameters τ^p_{abs}, τ^s_{abs}, $2\tau^p_{dif}$, $\tau^s_{dif} \in (0,1]$. MSC^+ sequences with the same length (l) whose respective MSCs are all strongly disambiguated to the same PoS class and the same sense, are considered occurrences (instances) of the same MSC^+ pattern, as stated by Definitions 5 and 7.

Definition 5. MSC^+ ***pattern***. *An MSC^+ pattern is an ordered list of $l \geq 1$ pairs $pmsc^+ = (\langle p_1, s_1\rangle, \ldots, \langle p_l, s_l\rangle)$, with each pair $\langle p_k, s_k\rangle$ ($1 \leq k \leq l$) referring to a PoS class p_k and a sense s_k.*

Definition 6. ***Perfectly matching instance of*** MSC^+ ***pattern***. *Given an MSC^+ pattern $pmsc^+ = (\langle p_1, s_1\rangle, \ldots, \langle p_l, s_l\rangle)$ with length $l \geq 1$ a perfectly matching instance of pmsc is any MSC^+ sequence $msc^+ = (msc_1, \ldots, msc_l)$ also of length l, such that each morpho-semantic component $msc_k = \langle idd_k, idc_k, b_k, e_k, p_k, s_k\rangle$ ($1 \leq k \leq l$) of msc^+ is strongly disambiguated, according to the parameters τ^p_{abs}, τ^s_{abs}, τ^p_{dif}, $\tau^s_{dif} \in (0,1]$, to the respective morphosyntactic class p_k and sense s_k (pair $\langle p_k, s_k\rangle$) of $pmsc^+$.*

Notice that distinct instances of an MSC^+ pattern can appear in distinct documents. It has been exemplified in the tweets of Table 2, each one with an instance of the pattern $\langle Verb, Ingest\rangle$, $\langle Adj, Any\rangle$, $\langle Noun, Beer\rangle$.

Finally, we say that an MSC sequence MSC^+ of a text document d partially matches an MSC pattern $pmsc^+$ when at least one component msc_k of that MSC^+ sequence matches the sense and the morphosyntactic class of the respective pair $\langle p_k, s_k\rangle$ of the pattern $pmsc^+$ as stated by Definition 7. The senses of the remaing MSC components of msc^+ are unsolved, i.e., undefined (with no candidate sense) or at least not strongly disambiguated. Though this definition allows partially matching instances with any number of MSC components with sense unsolved, the experiments reported in this paper only considered partially matching instances with just one component with sense undefined, for doing WSI/WSD to solve the sense of the respective word.

Definition 7. ***Partially matching instance of MSC$^+$ pattern.*** *Given an MSC^+ pattern $pmsc^+ = (\langle p_1, s_1\rangle, \ldots, \langle p_l, s_l\rangle)$ with length $l \geq 1$ and an MSC^+ sequence $msc^+ = (msc_1, \ldots, msc_l)$ also of length l, we say that MSC^+ partially matches the MSC pattern $pmsc$ if at least 1 MSC component of MSC^+ is strongly disambiguated, according to the parameters τ^p_{abs}, τ^s_{abs}, τ^p_{dif}, $\tau^s_{dif} \in (0, 1]$, to the respective morphosyntactic class and sense of $pmsc^+$, i.e., $\langle p_k, s_k\rangle$, and at least one of the candidate morphosyntactic classes $p_{k,i} \in P_k$ $(1 \leq k \leq l, i \geq 1)$ of each MSC in MSC^+ matches the one of the respective pair $\langle p_k, s_k\rangle$ of the pattern $pmsc$.*

5. The Proposed Approach

This section describes our approach for coping with challenging instances of the WSI and WSD problems in sequences of texts such as social media posts. Figure 1 provides an overview of the proposed process, which is composed of five stages: (i) Pre-processing, (ii) Semantic and morphosyntactic annotation, (iii) MSC extraction, (iv) Matching MSCs & Mining MSC^+ patterns, and (v) WSI/WSD based on MSC^+ patterns. In this figure, the continuous lines indicate data flow and the dotted lines linking stage (ii) with morphosyntactic annotation tools (e.g., FreeLing, LX-Tagger) and WSI/WSD and/or NER/NED tools (e.g., DBpedia-Spotlight, Babelfy) indicate its function dependence on the use of such tools, which can be accessed remotely by web servers via APIs. Many of the latter rely on sense inventories, such as lexicons (e.g. WordNet), Linked Open Data (LOD) collections (e.g. DBpedia) or compositions of them (e.g. Babelnet), usually represented as knowledge graphs.

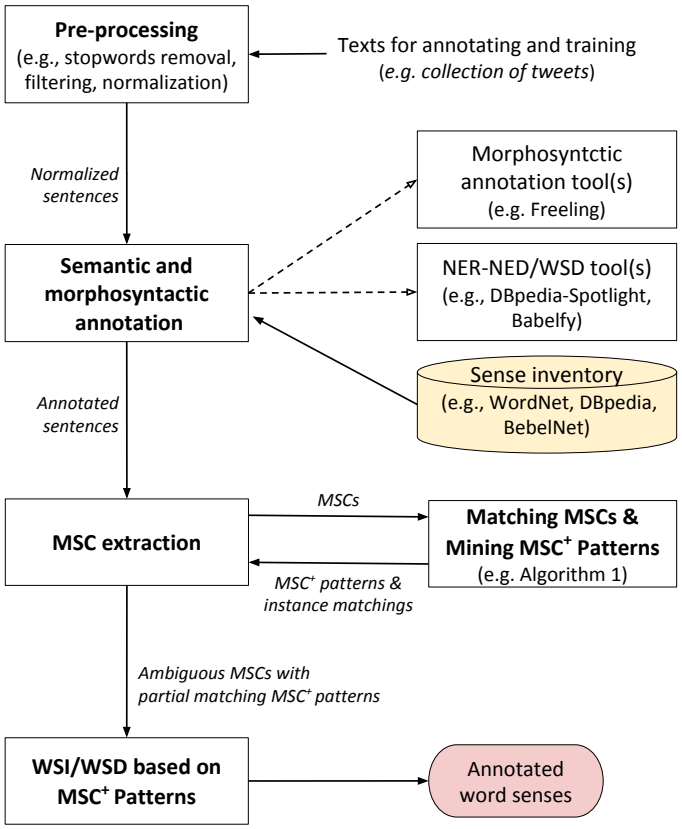

Figure 1: Proposed approach.

5.1. Pre-processing

Short texts such as social media posts usually have few and sparse context data. However, we can select collection of posts of certain *users*, with some *hashtag*, originated from a certain region and/or whose *timestamp* is in a certain time interval. The texts of such posts tend to present similar language habits and taken as collections

can provide more contextual information. Our method take advantage of this to extract language patterns that are common in such post collection and use them for WSI/WSD.

However, social media posts are very prone to noise. They may present *emoticons/emoji*, content in different languages, bad use of capitalization, lack of vowels/consonants in abbreviation and acronyms, slangs, characters not recognized by typical text *parsers* of annotation tools (e.g., @ = user, # = hashtag, url) and other problems. Therefore, first of all it is necessary to submit the posts to pre-processing task, for cleansing, filtering and normalization of their texts. Cleansing removes the less useful parts of the text, such as stopwords. Filtering, on the other side, can select the most relevant parts, based on keywords, topics, or language (e.g., English, Portuguese, Spanish). Normalization can exchange out-of-vocabulary words with equivalent formal ones from a repository.

In our work, we consider the useful parts of the text as being tokens that do add relevant information to the message such as verbs, nouns, adverbs, and adjectives. Table 4 provides an example of tweet raw text (Input) and how it becomes after the pre-processing stage (Output).

Table 4: An example of tweet text pre-processing.

Input:	RT @user: Star Wars #theforceawakens first reactions https://t.co/8KZno5CW49.
Output:	*user Star Wars theforceawakens reactions*

5.2. Semantic and morphosyntactic annotation

The semantic and morphosyntactic annotation enriches the relevant text components resulting from the pre-processing stage with PoS tags and semantic resources (e.g., concepts, instances, synsets) of knowledge graphs. Table 5 presents annotations generated by FreeLing[11] on relevant textual components resulting from the text pre-processing illustrated in Table 4. It associates to each token (on the left) its lemma (middle) and its morphosyntactical class (right). NP00V00 1 means that Freeling classified *Star_Wars* as a proper noun whose gender and number are unspecified and whose named entity class is *other* , with confidence 1 (100%). VBZ means that *theforceawakens* was classified as a verb in the third person with confidence 0.9967. Finally, NNS 1 means that *reactions* has been annotated as a common noun in plural with confidance 1. Freeling can also provide candidate meanings for some words, that we have not shown in Table 4 due to space limitation. For instance, among the possible senses for the word *reaction* Freeling lists the synset with id = 00859001-n[12] taken from the WordNet Multilingual Central Repository[13].

Table 5: Examples of annotated text components.

Star_Wars	star_wars	NP00V00 1
theforceawakens	theforceawakens	VBZ 0.9967
reactions	reaction	NNS 1

5.3. MSC extraction

After the previous stage, in which relevant words were annotated with morphosyntactic classes (e.g. PoS tags) and semantic resources (e.g., synsets of lexicons, LOD resources) that best match the textual context where they appear, our method extracts the relevant Morphosyntactic Components (MSCs). This is done based on the agreement and confidence of the annotations performed by state-of-the-art tools. Table 6 presents MSCs extracted from the annotated sentence presented in Table 5. Each one refers to *idc* and *idd*, an arbitrary word w and *lemma*, a list of candidate morphosyntactic classes P, and a list of candidate senses S, respectively. In particular, the MSC approach uses two types of con-

[11] http://nlp.lsi.upc.edu/freeling

[12] http://wordnet-rdf.princeton.edu/pwn30/00859001-n

[13] http://adimen.si.ehu.es/web/MCR/

Table 6: Examples of morpho-semantic components extracted from a tweet

word	MSC					
	idd	*idc*	*b*	*e*	*P*	*S*
Star_Wars	1	1	11	19	$\{\langle NPV, 1\rangle\}$	–
theforceawakens	1	2	22	36	$\{\langle VBZ, 0.9966\rangle, \ldots\}$	–
reactions	1	3	44	52	$\{\langle NNS, 1\rangle\}$	$\{\langle 859001 - n 0.0081\rangle, \ldots\}$

text word-features: semantic-based features and language-language features, described below.

Semantic-based features. We extract for each word a list of its candidate senses with their respective levels of confidence. This information is obtained by using annotation tools which implement NER/NED and/or WSI/WSD, usually based on local textual contexts. Relations between synsets also can be used for tagging senses. For instance, in WordNet the annotated synset of ID 859001-n (candidate sense for *idc* 3 in column S of Table 6), which belongs to the noun (n) morphological group, semantic domain *act*, defined by the synset {*reaction, response*} that means "*a bodily process occurring due to the effect of some antecedent stimulus or agent*". It is related to the synsets 02894436-a ({*sensorimotor*}) and 00717358-v ({*react, respond*}), via the relations "has_derived" and "retated_to", respectively.

Language-based features. These features are based on n-gram probabilities. In particular, the context of a word w is represented by (i) one or more neighboring words on its left and its right (e.g. "*theforceawakens reaction*", "*star_wars reaction*", "*star_wars theforceawakens reaction*"); and (ii) their respective lists of candidate pairs $\langle P, S\rangle$ of morphosyntactic class P and sense S, $\{\langle VBZ, -\rangle, \langle NNS, 859001\text{-}n\rangle\}$, $\{\langle NPV, -\rangle, \langle NNS, 859001\text{-}n\rangle\}$, and $\{\langle NPV, -\rangle, \langle VBZ, -\rangle, \langle NNS, 859001\text{-}n\rangle\}$. The n-gram probabilities are also used as belief values for PoS tagging and sense candidates.

5.4. Matchig MSCs and Mining MSC^+ patterns

Algorithm 1 depicts our method for mining MSC^+ patterns and associating each one of them with its respective (partial) matching instances. It takes as inputs Morpho-Semantic Components (MSCs) extracted from a set of text documents and the minimum support (number of instances) $sup \geq 1$ for returned mined patterns.

Algorithm 1: Mining MSC^+ patterns

```
input  : mscData; // list of MSCs found in a
         set of text documents
input  : sup; // minimum support
output: MSC+ patterns with their instances.
1  begin
2      foreach p ∈ P do
3          foreach s ∈ S do
4              matchings ← filter msc ∈
                mscData matching ⟨p, s⟩;
5              if matchings.size ≥ sup then
6                  matchingMSCs[p, s] ←matchings;
7      mscMines(mscData, matchingMSCs, sup);
8  mscMines (mscData, matchingMSCs, sup):
9      foreach k, v ∈ mscData do
10         foreach i, c ∈ matchingMSCs do
11             if k ≠ i then
12                 mscIntersection ← v ∩ c;
13                 if mscIntersection ≠ ∅ then
14                     mscMine[k, i] ←
                        mscIntersection;
15     if mscMine ≠ ∅ then
16         fileOut(mscMine);
17         mscMines(msc, mscMine, sup, n);
```

The first call of this algorithm takes all MSCs with resolved senses from a training data set. Afterwards, this algorithm can be called for finding matches for MSCs (usually from new documents) whose sense has to be induced and/or disambiguated. MSCs in the sentences have been automatically extracted as explained before, and

labeled in vertical data format (Fournier-Viger et al., 2017). MSCs having compatible morphosyntactic classes and senses are considered to match each other. A partial matching occurs when some document one or more MSCs that match a pattern, being one of these MSCs with the sense not defined or not disambiguated.

First (lines 1 to 6 of Algorithm 1), we analyze the incidences of combinations between morphosyntactic tags and senses. Patterns with length one (of the form $\{\langle p, s\rangle\}$) are generated by pruning based on support *sup*. Their instances (with one MSC each one) are stored in the vector $interactingMSC$. Then, we mine patterns (function $mscMines$) of length greater than one in a recursive way, by comparing the content and ids of canditates *msc* ($v \cap c$, with left alignment vector). Thus, a sequence of components MSC^+ has high relevance when it is suportted by several instances whose morphosyntactic class and sense are compatible with those of a pair $\langle p, s\rangle$. This function also implements pruning based on support values, filters to select the content and verify the order of the components.

MSC^+ patterns and the respective instances resulting from Algorithm 1 are returned in a file in JSON format. Table 2 presents two perfectly matching instances of the pattern of size three: $\{\langle Verb, Ingest\rangle, \langle Adj, Any\rangle, \langle Noun, Beer\rangle\}$ occurring in the respective tweets with identifiers #1 and #2, respectively. This pattern can then be used for WSI and WSD as described in the following. As the other instances partially matching this pattern (tweets with identifiers between #3 and #5) have the sense of their last component disambiguated to *Beer* the pattern support increases.

5.5. WSI and WSD based on MSC^+ patterns

Algorithm 1 associates two kinds of instances (MSC sequences found in the text documents) to the respective MSC^+ patterns: (i) **perfectly matching instances**, i.e., MSC sequences in which *all components are strongly disambiguated* and each one matches the respective pair morphosyntctic class and sense $\langle p, s\rangle$ of some MSC^+ pattern as defined in Section 4; and (ii) **partially matching instances** i.e., MSC sequences in which *at least one of the components has its sense undefined, not disambiguated at all, or just weakly disambiguated*. If other components of a partially matching instance perfectly match the corresponding pair $\langle p, s\rangle$ of the MSC^+ pattern, the missing sense(s) can be induced and disambiguated in accordance with the sense of the respective $\langle p, s\rangle$ pair of the MSC^+ pattern.

For example, the sense of MSC_3 of tweets #3 (**polar**), #4 (**Devassa**) and #5 (**devassa**) of Table 3 can be disambiguated to **Beer** thanks to the partial matching of the respective MSC sequences to the pattern $\{\langle Verb, Ingest\rangle$, $\langle Adj, Any\rangle$, $\langle Noun, Beer\rangle\}$. We assume that the higher the support (number of MSC instances of the pattter, i.e., MSC sequences perfectly matching the pattern) and the lower the number of weakly disambiguated components in the partially matching MSC sequences, the highest the confidence for WSI/WSD.

An example of partially matching MSC sequence with more than one component with sense unsolved is the one given by the 3 component words "*star_wars*", "*theforceawakens*" and "*reaction*" in Table 6. Its first two words do not have their senses annotated (what is indicated by – sign in column *S*). In this case, each word with sense unresolved ("*star_wars*" and "*theforceawekens*" in the partially matching instance can be considered as a target words for the WSI and the WSD tasks.

Our approach for WSI and WSD consists of mining the MSC^+ patterns from a set of text documents, identifying pattern instances having at least one MSC with unresolved sense, and using the partial adherence to patterns for solving the sense of the respective words (with unsolved sense). WSI and WSD are done by using heuristics and statistical information from the patterns with sufficient support of perfectly matching MSC^+ sequence instances.

When a word whose sense is to be resolved is part of MSC^+ sequences partially matching more than one pattern, we use score methods for WSI/WSD. We consider matching such a word with the pairs morphosyntactic class and sense

($\langle p, s \rangle$) of the respective MSC in each pattern based on the beliefs $\tilde{p}$ and $\tilde{s}$ of the respective MSC components of each perfectly matching instance of each pattern. We experiment two score methods:

1. **Max Average Adherence for Sense (MAA-S)**: This method induces the sense of a target word w_t in an MSC^+ sequence (instance) partially matching more than one MSC^+ patterns by adherence to strongly disambiguated instances of these MSC^+ patterns. The MAA-S method computes the score for each pattern (and consequently the sense s of the corresponding MSC) by averanging the beliefs $\tilde{s}$ of the respective strongly disambiguated MSC of each perfectly matching instance of the pattern i in accordance with Equation 1.
2. **Max Average Adherence for morphosyntactic class and sense (MAA-PS)**: The MAA-PS method computes the WSI score using the beliefs $\tilde{p}$ and $\tilde{s}$ of each candidate morphosyntactic class p and sense s of the respective strongly disambiguated MSC (corresponding to the position of the word w_t) in each instance perfectly matching the pattern. The score of each w_t for each pattern i is computed in accordance with Equation 2.

$$Score(w_{ti}) = \frac{\sum_{j=1}^{sup_i} \tilde{s}_{ji}}{sup_i} \quad (1)$$

$$Score(w_{ti}) = \frac{\sum_{j=1}^{sup_i} (\tilde{p}_{ji} + \tilde{s}_{ji})}{2sup_i} \quad (2)$$

In Equations 1 and 2, sup_i is the total number of MSC^+ sequences perfectly matching an MSC^+ pattern i. Then, the sense for w_t is chosen from the corresponding position of the pattern with maximum score among the candidate ones ($BestFit$), as stated by Equation 3.

$$BestFit = \underset{i}{\operatorname{argmax}}\, Score(w_{ti}) \quad (3)$$

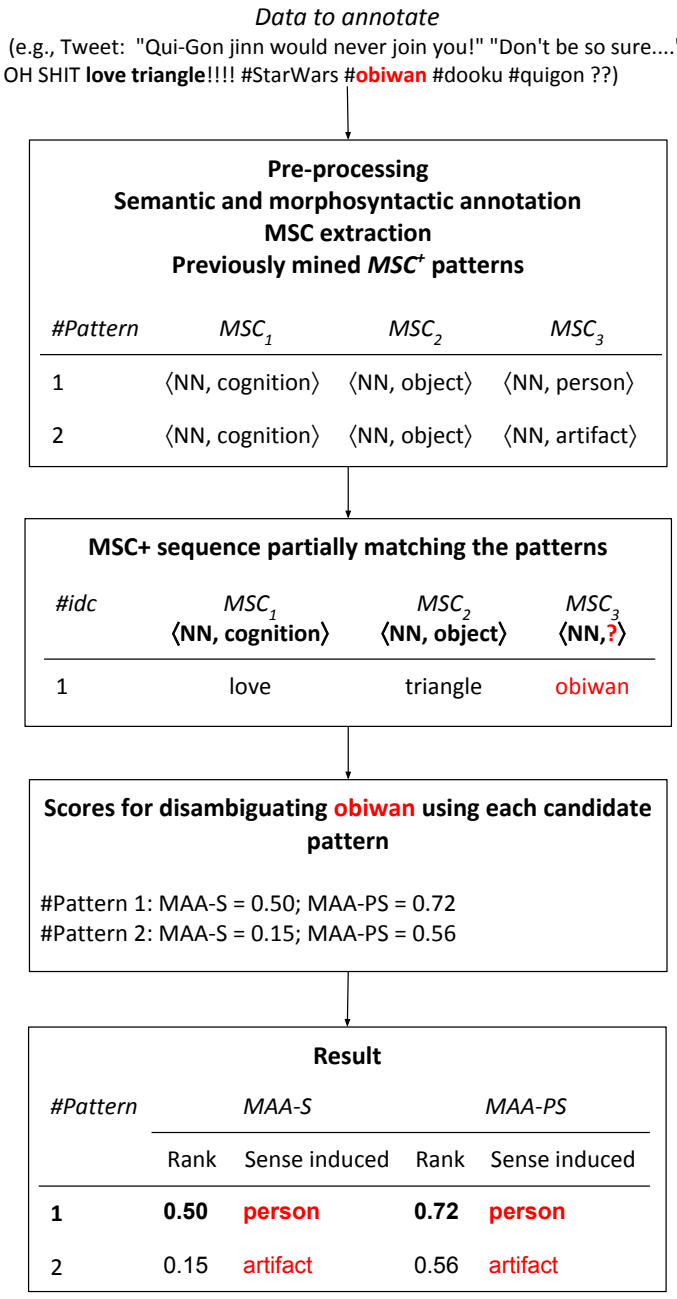

Figure 2: Example of WSI and WSD based on previously mined MSC^+ patterns on a target word of a tweet.

Figure 2 illustrates the application of our WSI/WSD approach (process shown in Figure 1) to a tweet with the MSC sequence (MSC_1, MSC_2, MSC_3), considering two previously mined MSC^+ patterns ($\#Pattern$ 1 and 2). First, the tweet tweet is pre-processed for eliminating noise, its words are annotated with morphosyntactic classes and senses by using off-the-shelf annotation tools, and each relevant MSC

is extracted from the tweet and submitted to Algorithm 1 to find matchings between MSC^+ sequences of the tweet and previously mined MSC^+ patterns. The second box of Figure 2 (from top to bottom) shows the words of the MSC sequence (MSC_1, MSC_2, MSC_3), with the target word *obiwan* (whose sense was not solved by the annotation tool) in red on the last position of the sequence (MSC_3). This MSC^+ sequence partially matches patterns 1 and 2. The sense for the target word *obiwan* (weakly disambiguated MSC_3) is taken from the corresponding position of pattern 1 (person), because this pattern has the highest score for the position corresponding to MSC_3 for both methods do calculate this score (MAA-S 0.50 and MAA-PS 0.72).

#Pattern 1:
MAA-S: (0.20 + 1.00 + 0.30) / 3 = 0.50
MAA-PS: (0.99 + 0.86 + 1.00 + 0.20 + 1.00 + 0.30) / (2 * 3) = 0.72

#Pattern 2:
MAA-S: (0.15 + 0.15) / 2 = 0.15
MAA-PS: (0.99 + 0.96 + 0.15 + 0.15) / (2 * 2) = 0.56

Beliefs of perfectly matching instances

#idc	#Pattern	MSC_1 $\langle\tilde{p}_1, \tilde{s}_1\rangle$	MSC_2 $\langle\tilde{p}_2, \tilde{s}_2\rangle$	MSC_3 $\langle\tilde{p}_3, \tilde{s}_3\rangle$
2	1	⟨1.00, 0.25⟩	⟨0.98, 0.50⟩	⟨0.99, 0.20⟩
3	1	⟨1.00, 0.25⟩	⟨0.98, 0.50⟩	⟨0.86, 1.00⟩
4	1	⟨0.98, 0.08⟩	⟨0.98, 0.08⟩	⟨1.00, 0.30⟩
5	2	⟨1.00, 0.25⟩	⟨0.98, 0.50⟩	⟨0.99, 0.15⟩
6	2	⟨1.00, 0.25⟩	⟨0.98, 0.50⟩	⟨0.96, 0.15⟩

Figure 3: Details of score calculation with the methods MAA-S and MAA-PS using instances perfectly matching each candidate pattern.

Figure 3 shows details of the calculus of these scores using the beliefs of each strong disambiguated MSC of the corresponding position (MSC_3) of each perfectly matching MSC^+ instance of each candidate pattern. MAA-S takes the average of the beliefs on sense $(\tilde{s})$, while MAA-PS takes into account both kinds of of beliefs: one sense ($\tilde{s}$, in blue) and on morpho-syntactic class ($\tilde{p}$, in green).

6. Experiments

The experiments aim to investigate the prevalence of MSC^+ patterns in tweets, their support by perfectly matching MSC^+ sequences in these tweets, and the gains obtained by doing WSI/WSD based on the mined patterns on words with sense unresolved by typical annotation tools. This section describes the infra-structure, including off-the-shelf tools, the parameters, the dataset, and the evaluation metrics used in these experiments. It also presents a characterization of the mined patterns and their matching instances, as well as the time spent to mine them. The results regarding WSI/WSD with the mined patterns are reported and discussed in Section 7.

6.1. Tools and parameters

The support parameter of Algorithm 1 was set to 2% and 5% of the number of training tweets. Note that no additional parameter tuning is needed for our approach. After mining the MSC^+ patterns, we selected the ones with length 3 for evaluation. We evaluated our MAA-S and MAA-PS methods for WSI/WSD by assessing the semantic annotation improvements (precision and recall) over the results of two annotation tools:

FreeLing. This tool provides language analysis functionalities useful to construct MSC, such as text tokenization, morphological analysis, named entity detection and classification, PoS tagging, WordNet based sense annotation. FreeLing uses UKB, a state-of-the-art WSD system based on Personalized PageRank (Agirre and Soroa, 2009).

Babelfy. This state-of-the-art toll for WSD and entity linking on text written in variety of languages draws upon BabelNet 3.0, a large multilingual semantic network which connects descriptions of concepts and objects from different inventories, such as WordNet, Open Multilingual WordNet, Wikipedia, OmegaWiki, Wiktionary and Wikidata (Navigli and Ponzetto, 2010).

FreeLing&Babelfy annotation is a composition of the results generated by FreeLing PoS tagging and Babelfy sense annotation.

6.2. Dataset

We used the Microposts2016[14] dataset for testing different configurations for the proposed WSI/WSD approach. Microposts2016 consists of event and non-event tweets extracted from a collection of over 18 million ones. After removing from Microposts2016 the tweets "Not Available" (not in the Twitter server anymore) we obtained 2493 tweets, with 827 annotated mentions to entities in the gold standard. The annotation of these tweets using FreeLing&Babelfy allowed us to extract 16260 relevant strongly disambiguated $MSCs$ from these tweets, since 2,6K extracted words were weakly disambiguated or with no candidate sense annotated by Babelfy.

6.3. Evaluation metrics

Micropost2016 provides a gold standard to entities corresponding to the hierarchy of the semantic domain of words from WordNet (also called Supersenses). Supersenses are coarse-grained semantic labels based on syntactic categories (nouns, verbs, adjectives and adverbs) and logical groupings of senses in classes such as person, phenomenon, feeling, and location (45 groups in total) (Vu et al., 2017). Therefore, to measure WSI/WSD performance, we used the supersense to induce and disambiguate senses, as the task *one supersense per collocation* (analogue of one sense per collocation), and mapped them to the gold standard. To evaluate the performance of the proposed WSI/WSD approach, we used the classical precision, recall an F-score measures based on the matching of the DBpedia class with the supersense class returned by the annotation tool.

6.4. Mined patterns

Table 7 summarizes the results (number of MSC^+ patterns with length between 1 and 5, and their perfectly matching instances and partially matching instances) obtained by applying Algorithm 1 to the MSCs extracted from the selected subset of Microposts2016 tweets. The required support of the mined patterns was set to 2% of the total number of tweets. Notice that the number of perfectly matching instances is higher than the number of partially matching instances. This situation usually occurs when the number of strongly disambiguated MSCs (discounting words with no sense annotated by the annotation tool) is higher than the the number of weakly disambiguated MSCs.

Table 7: Quantities of MSC^+ patterns and instances found in the dataset.

Length	MSC^+ patterns	Matching instances	
		Perfectly	Partially
1	108	22437	2675
2	1064	57882	14453
3	1943	45512	12726
4	562	5260	1474
5	9	47	9

6.5. Execution time

We measured the execution time of Algorithm 1 for mining MSC^+ patterns using a set of 20,000 MSCs extracted from the Microposts2016 training data. The set of MSCs was partitioned in four samples: 1,000 MSC (a-1K), 5,000 MSC (b-5K), 10,000 MSC (c-10K), and 15,000 MSC (d-15K).

Figure 4 presents the arithmetic average of the time (measured in seconds) spent on five executions of Algorithm 1 with each data sample. The measurements were done using a desktop with an Intel Core 2 Duo processor @3.00GHz, running Windows10, 64 bits, and routines invoked via command-line prompt.

As the size of the samples grows, the computation that needs to be performed by Algorithm 1 for mining MSC^+ patterns and matching their instances could end up being fairly substantial, due to the $O(n^2\ log\ n)$ time complexity. However, a simple partitioning scheme (e.g. extracting the important sentences) can help to speed up the algorithm when input sizes become large.

[14] `http://microposts2016.seas.upenn.edu/challenge.html`

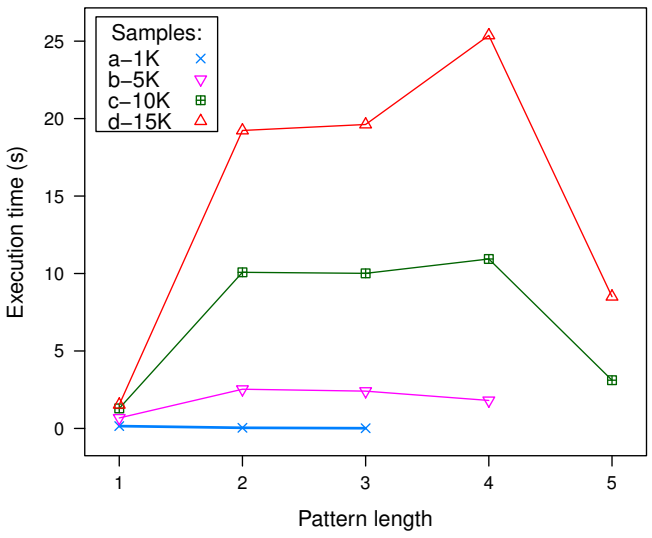

Figure 4: Execution times of the samples.

7. WSI/WSD results

Table 8 presents the WSI/WSD performance for FreeLing, FreeLing&Babelfy, and some variations of our proposal based on MSC^{+} patterns (below the dashed line). Tables 9 and Table 10, by their turn, detail the gains obtained by applying variations of our approach with support 2% and 5%, respectively. Each table shows the total number of words whose sense has been handled by the respective approach (#Word), the number of matching in terms of just surface name (Mention) or surface name and sense (Mention&sense), the recall (R), the precision (P), and the F-Measure (F). Notice that the columns R, P and F, in Tables 9 and Table 10 refer to the gains obtained in the respective measures by variation of our method, which solved the sense of a number of words whose sense was left undefined (with no sense found) by the baseline systems.

Table 8 highlights in bold the best results for recall (R), precision (P) and F-score (F) for each system evaluated. The patterns with the support of 2% (MSC2%) achieved the best results for recall improving the annotation of the systems in 8.9% to FreeLing (0.709 to 0.798) and 3.5% to FreeLing&Babelfy (0.712 to 0.747).

The proposed WSI method MAA-PS achieved the best results for precision in both systems and support of patterns (MSC2% and MSC5%). In FreeLing system the precision of 0.210 and F-score of 0.324 could be improved to 0.225 and 0.345, respectively using FreeLing+MAA-PS (MSC5%). In FreeLing&Babefy system the precision of 0.238 and F-score of 0.357 could be improved to 0.243 and 0.364, respectively using FreeLing&Babefy+MAA-PS (MSC5%). The FreeLing&Babefy+MAA-PS (MSC2%) achieved the best general result using our approach.

Table 8 also show the gain performance for the candidate target words (#Words) to the disambiguation task. Using the FreeLing and FreeLing&Babelfy systems with MSC2%, our results show that gain greater than MSC5%, 1330 candidate target words to FreeLing and 685 to FreeLing&Babelfy, respectively.

Figure 5 graphically compares the results of each base baseline with those improved by using our MAA-PS method, in terms of recall (a) and precision (b). Considering MSC2% (supp = 2%) we can find more candidate instances for the WSI task than the support used is 5% (MSC5%), and consequently achieved the best recall results in the systems. In another hand, MSC5% achieved better results for precision than MSC2%. It is because if we increase the support value, matching MSC sequences will need to occur many times in the dataset to be considered a pattern, and consequently Algorithm 1 will find a smaller number of patterns.

MAA-S uses only word sense beliefs of instances strongly disambiguated while MAA-SP uses the PoS tagging and word sense beliefs. As we can see from Table 9 Table 10, the MAA-SP method allows achieved the best precision in both systems tested with different values of supports.

8. Conclusions and future work

This paper presented an approach for WSI/WSD that exploits language patterns composed by sequences of morpho-semantic components (MSC) quite frequent in some short text documents such as tweets. We presented an algorithm to mine these patterns, which we call MSC^{+} patterns, from sets of text documents. It

Table 8: WSI/WSD performance

System	#Words	Matching		R	P	F
		Mention	Mention&sense			
FreeLing	13594	586	174	0.709	0.210	0.324
FreeLing&Babelfy	13770	589	197	0.712	0.238	0.357
FreeLing+MAA-PS (MSC2%)	14924	660	179	**0.798**	0.216	0.287
FreeLing&Babelfy+MAA-PS (MSC2%)	14455	618	200	**0.747**	0.242	0.365
FreeLing+MAA-PS (MSC5%)	14226	613	187	0.741	**0.225**	0.345
FreeLing&Babelfy+MAA-PS (MSC5%)	14046	596	201	0.721	**0.243**	0.364

Table 9: MSC^{+} pattern performance with support = 2% (MSC2%)

System	Method	#Words	Matching		R	P	F
			Mention	Mention&sense			
FreeLing	MAA-S	1330	74	4	0.089	0.005	0.009
	MAA-PS	1330	74	5	0.089	0.006	0.011
FreeLing&Babelfy	MAA-S	685	29	1	0.035	0.001	0.002
	MAA-PS	685	29	3	0.035	0.004	0.007

Table 10: MSC^{+} pattern performance with support = 5% (MSC5%)

System	Method	#Words	Matching		R	P	F
			Mention	Mention&sense			
FreeLing	MAA-S	632	27	11	0.033	0.013	0.019
	MAA-PS	632	27	12	0.033	0.015	0.020
FreeLing&Babelfy	MAA-S	227	7	1	0.080	0.001	0.002
	MAA-PS	227	7	4	0.080	0.005	0.006

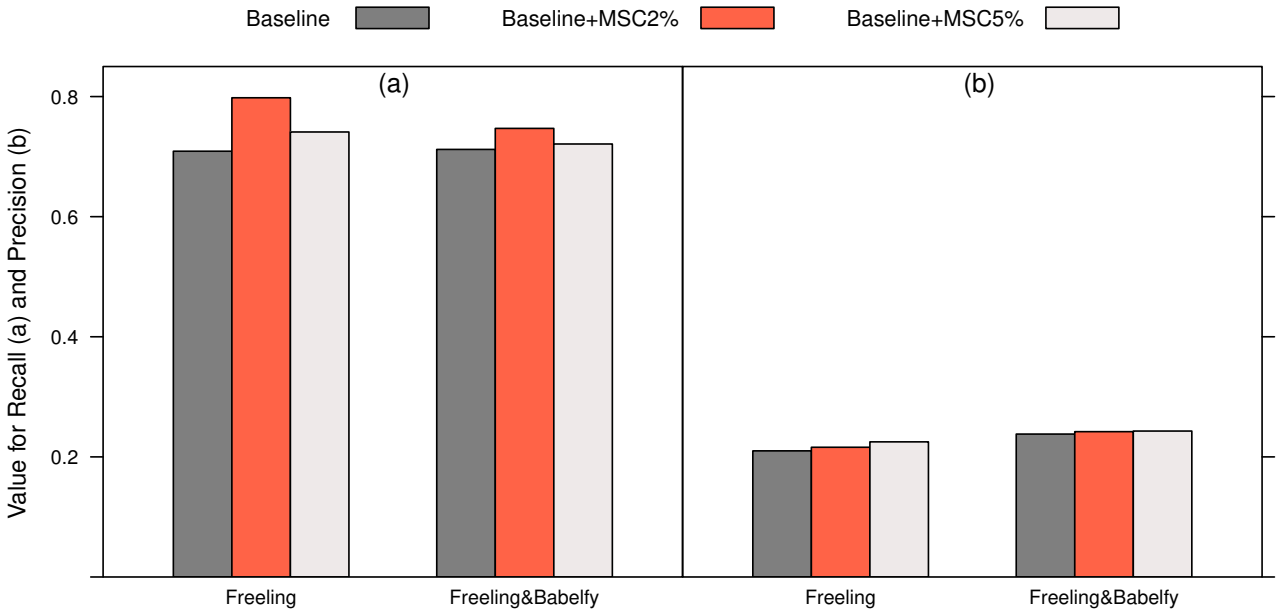

Figure 5: Comparison of recall (a) and precision (b).

relies on annotations of morphosyntactic classes and senses of relevant words. These annotations can be produced by a variety of alternative off-the-shelf tools for NLP. Our MSC^+ pattern algorithm can be seen as a kind of automatic learning of linguistic patterns that carry contextual information that can be used to resolve the sense of certain words that appear in instances of these patterns with a sense that may not be useful for these works, but that is induced by the pattern. Our methods for WSI and WSD based on MSC^+ patterns can complement current approaches for lexical semantics resolution from both the conceptual and the empirical perspectives, even when context information is quite limited in particular documents.

Our experiments have shown that morphosyntactic classification and semantic annotations based on sense inventories and knowledge graphs can be used to gather language patterns that provide more general context information to complement that of specific documents to improve the performance of WSI/WSD on tweets. We also demonstrate that the confidence on distinct annotations (i.e. candidate morphosyntactic classes and senses of words) are both useful to improve WSD precision and recall. The experiments reported in this study are readily reproducible, as the algorithms for MSC^+ pattern mining and WSI/WSD methods based on these patterns are publicly available[15].

In future work, we plan to explore our approach and methods for WSI/WSD on more extensively experiments with other datasets and existing WSI and WSD systems. The proposed approach can be applied easily with other tools, sense inventories and semantic knowledge bases different from FreeLing and Babelfy, BabelNet and WordNet. Further evaluation of the proposal can use distinct datasets that do not involve social media texts, such as short text that can appear in some diaries and medical records. Furthermore, we intend to investigate a variety of alternatives to improve the performance of the proposed approach. For instance, we plan to associate global weights to each MSC^+ pattern mined by our algorithm to improve the disambiguation process. We also intend to investigate the effects of several words with sense unsolved in the same MSC sequence on WSI/WSD, and smarter ways to incorporate the influence of newly disambiguated words in the suport of MSC^+ patterns and beliefs on particular morphosyntactic classes and senses.

[15] `https://github.com/fabiobif/MSC-patterns`

9. Acknowledgments

This work was conducted during a doctorate supported by grants of CAPES (Brazilian Coordination of Superior Level Staff Improvement) a research support agency from the Ministry of Education of Brazil. CAPES also supported an internship for international cooperation with the TALN (Natural Language Processing Research Group) at the Pompeu Fabra University in Barcelona, Spain.

References

A. Júnior, J.G., Schiel, U., Marinho, L.B., 2015. An approach for building lexical-semantic resources based on heterogeneous information sources, in: Proceedings of the 30th Annual ACM Symposium on Applied Computing, ACM. pp. 402–408.

Agirre, E., Martínez, D., de Lacalle, O.L., Soroa, A., 2006. Two graph-based algorithms for state-of-the-art wsd, in: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics. pp. 585–593.

Agirre, E., Soroa, A., 2009. Personalizing pagerank for word sense disambiguation, in: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics. pp. 33–41.

Al-Harbi, O., Jusoh, S., Norwawi, N.M., 2017. Lexical disambiguation in natural language questions (nlqs). arXiv preprint arXiv:1709.09250 .

Alagić, D., Šnajder, J., Padó, S., 2018. Leveraging lexical substitutes for unsupervised word sense induction, in: Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18).

Albano, L., Beneventano, D., Bergamaschi, S., 2014. Word sense induction with multilingual features representation, in: Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 02, IEEE Computer Society. pp. 343–349.

Artiles, J., Amigó, E., Gonzalo, J., 2009. The role of named entities in web people search, in: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2, Association for Computational Linguistics. pp. 534–542.

Bashar, M.A., Li, Y., Shen, Y., Gao, Y., Huang, W., 2017. Conceptual annotation of text patterns. Computational Intelligence 33, 948–979.

Béchet, N., Cellier, P., Charnois, T., Crémilleux, B., 2012. Discovering linguistic patterns using sequence mining, in: International Conference on Intelligent Text Processing and Computational Linguistics, Springer. pp. 154–165.

Biemann, C., 2006. Chinese whispers: an efficient graph clustering algorithm and its application to natural language processing problems, in: Proceedings of the first workshop on graph based methods for natural language processing, Association for Computational Linguistics. pp. 73–80.

Bontcheva, K., Derczynski, L., 2016. Chapter 6 - extracting information from social media with gate, in: Tonkin, E.L., Tourte, G.J. (Eds.), Working with Text. Chandos Publishing. Chandos Information Professional Series, pp. 133 – 158. URL: http://www.sciencedirect.com/science/article/pii/B9781843347491000068, doi:https://doi.org/10.1016/B978-1-84334-749-1.00006-8.

Brody, S., Lapata, M., 2009. Bayesian word sense induction, in: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics. pp. 103–111.

Camacho-Collados, J., Pilehvar, M.T., Navigli, R., 2016. Nasari: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. Artificial Intelligence 240, 36–64.

Chang, H.S., Agrawal, A., Ganesh, A., Desai, A., Mathur, V., Hough, A., McCallum, A., 2018. Efficient graph-based word sense induction by distributional inclusion vector embeddings. arXiv preprint arXiv:1804.03257 .

Di Marco, A., Navigli, R., 2013. Clustering and diversifying web search results with graph-based word sense induction. Computational Linguistics 39, 709–754.

Dragoni, M., 2018. Computational advertising in social networks: an opinion mining-based approach, in: Proceedings of the 33rd Annual ACM Symposium on Applied Computing, ACM. pp. 1798–1804.

Fellbaum, C., 1998. WordNet. Wiley Online Library.

Fileto, R., May, C., Renso, C., Pelekis, N., Klein, D., Theodoridis, Y., 2015. The baquara 2 knowledge-based framework for semantic enrichment and analysis of movement data. Data & Knowledge Engineering 98, 104–122.

Fournier-Viger, P., Lin, J.C.W., Kiran, R.U., Koh, Y.S., Thomas, R., 2017. A survey of sequential pattern mining. Data Science and Pattern Recognition 1, 54–77.

Goularte, F.B., Nassar, S.M., Fileto, R., Saggion, H., 2019. A text summarization method based on fuzzy rules and applicable to automated assessment. Expert Systems with Applications 115, 264–275.

Gutiérrez, Y., Vázquez, S., Montoyo, A., 2017. Spreading semantic information by word sense disambiguation. Knowledge-Based Systems 132, 47–61.

Jurafsky, D., Martin, J.H., 2018. Speech and language processing. 3nd ed. Draft chapters in progress, August 28, 2017.

Jurgens, D., Klapaftis, I., 2013. Semeval-2013 task 13: Word sense induction for graded and non-graded senses, in: Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), pp. 290–299.

Lau, J.H., Cook, P., McCarthy, D., Newman, D., Baldwin, T., 2012. Word sense induction for novel sense detection, in: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics. pp. 591–601.

Li, J., Jurafsky, D., 2015. Do multi-sense embeddings improve natural language understanding? arXiv preprint arXiv:1506.01070 .

Lin, D., 1998. Automatic retrieval and clustering of similar words, in: Proceedings of the 17th international conference on Computational linguistics-Volume 2, Association for Computational Linguistics. pp. 768–774.

Lin, D., Pantel, P., 2001. Dirt@ sbt@ discovery of inference rules from text, in: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, ACM. pp. 323–328.

Liu, P.Y., Liu, S., Yu, S.W., Zhao, T.J., 2010. Pengyuan@ pku: Extracting infrequent sense instance with the same n-gram pattern for the semeval-2010 task 15, in: Proceedings of the 5th International Workshop on Semantic Evaluation, Association for Computational Linguistics. pp. 371–374.

Liu, X., Zhou, M., Wei, F., Fu, Z., Zhou, X., 2012. Joint inference of named entity recognition and normalization for tweets, in: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, Association for Computational Linguistics. pp. 526–535.

Mahdisoltani, F., Biega, J., Suchanek, F.M., 2013. Yago3: A knowledge base from multilingual wikipedias, in: CIDR.

Mihalcea, R.F., 2002. Word sense disambiguation with pattern learning and automatic feature selection. Natural Language Engineering 8, 343–358.

Moro, A., Navigli, R., 2015. Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking, in: Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015), pp. 288–297.

Moro, A., Raganato, A., Navigli, R., 2014. Entity linking

meets word sense disambiguation: a unified approach. Transactions of the Association for Computational Linguistics 2, 231–244.

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

Navigli, R., 2012. A quick tour of word sense disambiguation, induction and related approaches, in: International Conference on Current Trends in Theory and Practice of Computer Science, Springer. pp. 115–129.

Navigli, R., Ponzetto, S.P., 2010. Babelnet: Building a very large multilingual semantic network, in: Proceedings of the 48th annual meeting of the association for computational linguistics, Association for Computational Linguistics. pp. 216–225.

Navigli, R., Vannella, D., 2013. Semeval-2013 task 11: Word sense induction and disambiguation within an end-user application, in: Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), pp. 193–201.

Panchenko, A., Ruppert, E., Faralli, S., Ponzetto, S.P., Biemann, C., 2017. Unsupervised does not mean uninterpretable: The case for word sense induction and disambiguation, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pp. 86–98.

Pelevina, M., Arefyev, N., Biemann, C., Panchenko, A., 2017. Making sense of word embeddings. arXiv preprint arXiv:1708.03390 .

Rodriguez, M.Z., Comin, C.H., Casanova, D., Bruno, O.M., Amancio, D.R., Rodrigues, F.A., Costa, L.d.F., 2016. Clustering algorithms: A comparative approach. arXiv preprint arXiv:1612.08388 .

Rosenberg, A., Hirschberg, J., 2007. V-measure: A conditional entropy-based external cluster evaluation measure, in: Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL).

Ruiz-Casado, M., Alfonseca, E., Castells, P., 2007. Automatising the learning of lexical patterns: An application to the enrichment of wordnet by extracting semantic relationships from wikipedia. Data & Knowledge Engineering 61, 484–499.

Saggion, H., Štajner, S., Bott, S., Mille, S., Rello, L., Drndarevic, B., 2015. Making it simplext: Implementation and evaluation of a text simplification system for spanish. ACM Transactions on Accessible Computing (TACCESS) 6, 14.

Schneider, N., Hovy, D., Johannsen, A., Carpuat, M., 2016. Semeval-2016 task 10: Detecting minimal semantic units and their meanings (dimsum), in: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), pp. 546–559.

Schütze, H., 1998. Automatic word sense discrimination. Computational linguistics 24, 97–123.

Strapparava, C., Gliozzo, A., Giuliano, C., 2004. Pattern abstraction and term similarity for word sense disambiguation: Irst at senseval-3, in: Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text.

Véronis, J., 2004. Hyperlex: lexical cartography for information retrieval. Computer Speech & Language 18, 223–252.

Vu, X.S., Flekova, L., Jiang, L., Gurevych, I., 2017. Lexical-semantic resources: yet powerful resources for automatic personality classification. arXiv preprint arXiv:1711.09824 .

Wu, Z., Li, C., Zhao, Z., Wu, F., Mei, Q., 2018. Identify shifts of word semantics through bayesian surprise, in: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, ACM. pp. 825–834.

Xia, C., Hu, J., Zhu, Y., Naaman, M., 2015. What is new in our city? a framework for event extraction using social media posts, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer. pp. 16–32.

Zhang, W., Su, J., Tan, C.L., Wang, W.T., 2010. Entity linking leveraging: Automatically generated annotation, in: Proceedings of the 23rd International Conference on Computational Linguistics, Association for Computational Linguistics, Stroudsburg, PA, USA. pp. 1290–1298. URL: http://dl.acm.org/citation.cfm?id=1873781.1873926.