

Contents lists available at [SciVerse ScienceDirect](#)

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Inferring the semantic properties of sentences by mining syntactic parse trees

Boris A. Galitsky^{a,*}, Josep Lluís de la Rosa^b, Gábor Dobrocsi^b

^a eBay Inc., San Jose, CA 95125, USA

^b EASY Innovation Center, Campus Montilivi, Univ. Girona, Catalonia, Spain

ARTICLE INFO

Article history:

Received 11 May 2010

Received in revised form 22 July 2012

Accepted 22 July 2012

Available online xxxx

Keywords:

Machine learning

Constituency parse tree

Search re-ranking

ABSTRACT

We extend the mechanism of logical generalization toward syntactic parse trees and attempt to detect semantic signals unobservable in the level of keywords. Generalization from a syntactic parse tree as a measure of syntactic similarity is defined by the obtained set of maximum common sub-trees and is performed at the level of paragraphs, sentences, phrases and individual words. We analyze the semantic features of this similarity measure and compare it with the semantics of traditional anti-unification of terms. Nearest-Neighbor machine learning is then applied to relate the sentence to a semantic class.

By using a syntactic parse tree-based similarity measure instead of the bag-of-words and keyword frequency approaches, we expect to detect a subtle difference between semantic classes that is otherwise unobservable. The proposed approach is evaluated in three distinct domains in which a lack of semantic information makes the classification of sentences rather difficult. We conclude that implicit indications of semantic classes can be extracted from syntactic structures.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Proceeding from parsing to the semantic level is an important step toward natural language understanding, with immediate applications in tasks such as information extraction and question answering [1,10,30,45]. Over the last decade, there has been a dramatic shift in computational linguistics from the manual construction of grammars and knowledge bases to partially or totally automating these processes using statistical learning methods trained on large annotated or un-annotated natural language corpora.

In this paper, we explore the possibility of high-level *semantic* classification of natural language sentences based on *syntactic(constituency) parse trees*. We address semantic classes appearing in information extraction (IE) and knowledge integration problems that usually require a deep natural-language understanding [6,8,12].

We attempt to combine the best of two worlds of linguistics and machine learning:

- 1) Rely on rich linguistic data such as constituency parse trees, and
- 2) Apply a systematic way to tackle this data, such as graph-oriented deterministic machine learning.

Notice that (1) gives us rather rich set of features compared to a bag-of-words approach or shallow parsing. We need to tackle such a rich set of features with its inherent structure by a structured machine learning approach. In this study we will evaluate how this richer set of tree-based features as a subject of graph-based learning outperforms keyword-based approaches in a number of text relevance problems.

* Corresponding author.

E-mail addresses: boris.galitsky@ebay.com (B.A. Galitsky), pepluis@silver.udg.edu (J.L. de la Rosa), gadomail@gmail.com (G. Dobrocsi).

Our approach is inspired by the notion of anti-unification [26,29] which is capable of generalizing arbitrary formulas in a formal language. We extend this notion towards anti-unifying of arbitrary linguistic structures such as constituency parse tree. In this paper we propose a definition and algorithm of a syntactic generalization which allows us to treat syntactic natural language expressions in a unified way as logic formulas.

Learning based on syntactic parse tree generalization is different from *kernel methods*, which are nonparametric density estimation techniques that compute a kernel function between data instances (which can include keywords, as well as their syntactic parameters), where a kernel function can be considered a similarity measure. Given a set of labeled instances, kernel methods determine the label of a novel instance by comparing it to the labeled training instances using the kernel function. Nearest neighbor classification and support-vector machines (SVMs) are two popular examples of kernel methods [62,63]. Compared to kernel methods, syntactic generalization can be considered structure-based and deterministic; linguistic features retain their structures and are not represented as values. Regarding the *edit distance* class of similarity methods, its analogue in anti-unification will be discussed in Section 6.1: such methods are better adjusted to objects with more peculiar structures, such as syntactic parse trees.

The main question considered in this study is whether these semantic patterns, unobservable at the level of keyword statistics, can be inferred from a complete parse tree structure. Moreover, the argumentative structures of the way authors communicate their conclusions (as expressed by their syntactic structures) are important in relating a sentence to the above classes. Studies [13,14] have demonstrated that graph-based machine learning can predict the plausibility of complaint scenarios based on their argumentation structures. Furthermore, we observed that learning the communicative structure of inter-human conflict scenarios can successfully classify the scenarios into a series of domains, from complaints to security-related domains. These findings convince us that applying a similar graph-based machine learning technique to such structures as syntactic trees, which have even weaker links to high-level semantic properties than these settings, can deliver satisfactory classification results. Graph based learning has been applied in a number of domains beyond linguistics (see e.g. [60]).

Most of the current learning research on NLP employs particular statistical techniques inspired by research on speech recognition, such as hidden Markov models (HMMs) and probabilistic context-free grammars (PCFGs). A variety of learning methods, including decision tree and rule induction, neural networks, instance-based methods, Bayesian network learning, inductive logic programming, explanation-based learning, and genetic algorithms can also be applied to natural-language problems and can present significant advantages in particular applications [25,46]. In addition to specific learning algorithms, a variety of general ideas from traditional machine learning, such as active learning, boosting, reinforcement learning, constructive induction, learning with background knowledge, theory refinement, experimental evaluation methods, and PAC learnability, may also be usefully applied to natural-language problems [10]. In this study, we employ the nearest neighbor type of learning, which is relatively simple, to focus our investigation on how expressive the similarity between syntactic structures can be in the detection of weak semantic signals. Other, more complex learning techniques can be applied, being more sensitive or more cautious, after we confirm that our measure of the syntactic similarity between texts is adequate.

The computational linguistics community has assembled large data sets for a range of interesting NLP problems. Some of these problems can be reduced to a standard classification task by appropriately constructing their features; however, others require using and/or producing complex data structures, such as complete parse trees and operations on these complete parse trees. In this paper, we introduce the generalization operation to a pair of parse trees for two sentences and demonstrate its role in sentence classification. The operation of generalization is defined starting at the level of lemmas and continuing through chunks/phrases all the way up to paragraphs/texts.

Learning syntactic parse trees allows one to conduct semantic inference in a domain-independent manner without using ontologies or other manually built resources. Training sets for text classification problems still need to be collected, but class assignment can be automated. Simultaneously, in contrast to most semantic inference projects, we will be restricted to a very specific semantic domain (limited set of classes), solving a number of practical problems.

The paper is organized as follows. We introduce three distinct problems of different complexities in which one or another semantic feature must be inferred from natural language sentences. We then describe the algorithm of the generalization of parse trees, followed by the nearest neighbor learning of the generalization results. The paper concludes with a comparative analysis of classification in selected problem domains, a search engine description, and a brief review of other studies with semantic inferences.

2. Application areas of syntactic generalization

In this study, we leverage the parse tree generalization technique in the automation of content management and a delivery platform [15,57] referred to as the Integrated Opinion Delivery Environment. This platform combines data mining of the web and social networks, content aggregation, reasoning, information extraction, questioning/answering and advertising to support distributed recommendation forums for a wide variety of products and services. In addition to human users, automated agents answer questions and provide recommendations based on previous postings by human users that are determined to be relevant. The key technological requirements are based on finding similarities between various types of texts. Therefore, the use of more complex structures to represent the texts' meanings is expected to improve the accuracy of the relevance assessment. Syntactic generalization has been deployed in content management and delivery platforms at two portals in Silicon Valley, USA: Datran.com and Zvents.com. We will present an evaluation of the way the accuracy of the relevance assessment has been improved in the Evaluation Section 6.

We focus on the three following problems, which are essential to various phases of the application described above:

1. detecting appropriate expressions for the automated building of ads as a component in an advertisement management platform;
2. classifying user postings with respect to their epistemic states: how well the user understands her product needs and how specific she currently is in her product choices;
3. classifying search results with respect to their relevance or irrelevance to search queries.

In all these tasks, it is necessary to relate a sentence into two classes, e.g. *suitable* vs. *unsuitable for particular purpose*, *knowledgeable* or *unknowledgeable user*, and *relevant/irrelevant answer* as the basis for ad generation. In both these tasks, decisions about membership in a class cannot be made when only the occurrence of specific words is provided; instead, peculiar and implicit linguistic information must be considered. It is difficult to formulate and even to imagine classification rules for both of these problems. However, finding plentiful examples for the respective classes is quite easy. We now outline each of these three problems.

As for the **first** problem of ad generation, its practical value is to assist business/website managers in writing ads for search engine marketing. Given the content of a website and its selected landing page, the system must select the most suitable sentences to form an ad.

As an example, consider the following content.

At Barclays, **we believe in great loan deals. That's why we offer a typical 9.9% APR on our loans of £7,500 to £25,000**.**

It's also why we pledge to pay the difference if you're offered a better deal elsewhere.

What you get with a personal loan from Barclays:

- * An instant decision if you're an Online Banking customer, and you **get your money in 3 hours**, if accepted†
- * Our price guarantee: if you're offered a better deal elsewhere, we'll pledge to pay you the difference between loan repayments***
- * **Apply to borrow up to £25,000**
- * No fees for arrangement or setup
- * Fixed monthly payments, so you know where you are
- * Optional tailored Payment Protection Insurance.

We want to generate ads as follows:

Great Loan Deals
9.9% APR typical on loans of
£7,500 to £25,000. Apply now!

Apply for a Barclays loan
We offer a typical 9.9% APR
Get your money in 3 hours!

We emphasize the sentences and their fragments for potential inclusion into an advert line (positive class) in bold. This is a semantic information extraction problem in which rules need to be formed automatically (a similar class of problem was formulated by Stevenson and Greenwood [31]). To form the criteria for an expression to be a candidate for an ad line, we apply the syntactic generalization to the sentences in the collected training sets and form templates from the generalization results, which are expected to be much more sensitive than the sets of keywords under the traditional keyword-based IE approach alone.

The **second** problem of the classification of epistemic states of a forum user is a more conventional classification problem, in which we determine what type of response a user is expecting:

- general recommendation,
- advice on a series of products, a brand, or a particular product,
- response and feedback on information shared, and others.

For each epistemic state, we have a training set of sentences, each of which is assigned to its state by a human expert. For example (epistemic states are italicized),

"I have no brand in particular in mind, but I have read that Canon makes good cameras" → *user with one brand in mind* or
"I have read a lot of reviews but still have some questions on what camera is right for me" → *experienced buyer*. We expect the proper epistemic state to be determined by the syntactically closest representative sentence.

Transitioning from keyword match to syntactic generalization is expected to significantly improve the accuracy of epistemic state classification because these states can be inferred from the syntactic structures of sentences rather than explicitly mentioned most of the time. Therefore, the results of the syntactic generalizations of the sentences that form the training set for each epistemic state will serve as classification templates, rather than the common keywords among these sentences.

The **third** evaluation of the matching mechanism is associated with the improvement of search relevance by measuring the similarity between queries and sentences in search results (or snapshots) by computing the syntactic generalization. This syntactic similarity is important when a search query contains keywords that form a phrase, domain-specific expression, or idiom, such as “shot-to-shot time” or “high number of shots in a short amount of time”. Usually, a search engine is unable to store all of these expressions because they are not necessarily frequent enough; however, they make sense if they occur within a certain natural-language expression.

The search implementation can be performed in two steps:

- 1) Keywords are formed from the query in a conventional manner, and search hits are obtained by considering the statistical parameters of the occurrences of these words in documents, popularity of hits, page rank and other factors.
- 2) The above hits are filtered with respect to the syntactic similarity of the snapshots of search hits with the search query. Parse tree generalization comes into play here.

Therefore, we obtain the results of the conventional search and calculate the score of the generalization results for the query, each sentence, and each search-hit snapshot. The search results are then re-sorted, and only those that are syntactically close to the search query are assumed to be relevant and returned to the user.

Let us consider an example of how to use the phrase-level match between a query and its candidate answer instead of a keyword-based match. When a query is relatively complex, it is important to perform the match at the phrase level instead of the keyword level (even considering the document popularity, TF*IDF, and the knowledge of which answers were previously selected by other users for similar queries).

For the following example www.google.com/search?q=how+to+pay+foreign+business+tax+if+I+live+in+the+US most of the search results are irrelevant. However, once one considers the syntactic structure of the query phrases, such as ‘pay-foreign-business-tax’ or ‘I-live-in-US’, irrelevant answers in which the keywords co-occur in a different way than they appear in the query are filtered out.

3. Generalizing portions of text

To measure the similarity of abstract entities expressed by logic formulas, a least-general generalization was proposed for a number of machine learning approaches, including explanation-based learning and inductive logic programming. Least-general generalization was originally introduced by Plotkin [26]. It is the opposite of most-general unification [27]; therefore, it is also known as *anti-unification*. Anti-unification was first studied in Plotkin and Robinson [26,27]. As its name suggests, given two terms, it produces a more general term that covers both, rather than a more specific one, as in unification. Let E_1 and E_2 be two terms. Term E is a generalization of E_1 and E_2 if there exist two substitutions σ_1 and σ_2 such that $\sigma_1(E) = E_1$ and $\sigma_2(E) = E_2$. The most specific generalization of E_1 and E_2 is called the anti-unifier. Here, we apply this abstraction to anti-unify such data as texts that are traditionally referred to as unstructured.

In this study, to measure the similarity between portions of text such as paragraphs, sentences and phrases, we extend the notion of generalization from logic formulas to the sets of syntactic parse trees of these portions of text. If it were possible to define the similarities between natural language expressions at a purely semantic level, the least-general generalization would be sufficient. However, in horizontal search domains where the construction of full ontologies for a complete translation from NL to logic language is not plausible, an extension of the abstract operation of generalization to the syntactic level is required. Rather than extracting common keywords, the generalization operation produces a syntactic expression that can be semantically interpreted as a common meaning shared by two sentences.

Let us represent a meaning of two NL expressions using logic formulas and then construct the unification and anti-unification of these formulas. How can we express a commonality between the expressions?

- *camera with digital zoom*
- *camera with zoom for beginners*

To express the meanings, we use the predicates *camera(name_of_feature, type_of_users)* (in real life, we would have a much higher number of arguments), and *zoom(type_of_zoom)*. The above NL expressions will be represented as follows:

```
camera(zoom(digital), AnyUser)
camera(zoom(AnyZoom), beginner),
```

where the variables (uninstantiated values that are not specified in NL expressions) are capitalized. Given the above pair of formulas, unification computes their most general specialization *camera(zoom(digital), beginner)*, and anti-unification computes their most specific generalization, *camera(zoom(AnyZoom), AnyUser)*.

At the syntactic level, we have the generalization of two noun phrases as follows:

```
{NN-camera, PRP-with, [digital], NN-zoom [for beginners]}.
```

We eliminate the expressions in square brackets because they occur in one expression and do not occur in another. Thus, we obtain {*NN-camera*, *PRP-with*, *NN-zoom*}, which is a syntactic analog to the semantic generalization above.

Because the constituent trees keep the sentence order intact, building structures upward to form phrases, we select a constituent tree to introduce our phrase-based generalization algorithm. The dependency tree has word nodes at different levels, and each word modifies another word or the root. Because it does not introduce phrase structures, the dependency tree has fewer nodes than the constituent tree and is less suitable for generalization. The constituent tree explicitly contains the word alignment-related information required for generalization at the level of phrases. We use the OpenNLP [53] system to derive the constituent trees for generalization (chunker and parser). Dependency-tree based, or graph-based, similarity measurement algorithms [13,59] are expected to perform as well as the one we focus on in this paper.

3.1. Generalizing at various levels: from words to paragraphs

The purpose of an abstract generalization is to find the commonality between portions of text at various semantic levels. The generalization operation occurs on the following levels:

- Article
- Paragraph
- Sentence
- Phrases (noun, verb and others)
- Individual word

At each level except the lowest one, individual words, the result of the generalization of two expressions is a *set* of expressions. In such a set, expressions for which less-general expressions exist are eliminated. The generalization of two sets of expressions is a set of the sets that are the results of the pair-wise generalization of these expressions.

We first outline the algorithm for two sentences and then proceed to the specifics for particular levels. The algorithm we present in this paper concerns paths of syntactic trees rather than sub-trees because these paths are tightly connected with language phrases. Regarding the operations on trees, we follow the work of Kapoor and Ramesh [48].

Although it is a formal operation on abstract trees, the generalization operation yields semantic information about the commonalities between sentences. Rather than extracting common keywords, the generalization operation produces a syntactic expression that can be semantically interpreted as a common meaning shared by two sentences.

- 1) Obtain the parsing tree for each sentence. For each word (tree node), we have a lemma, a part of speech and the form of the word's information. This information is contained in the node label. We also have an arc to the other node.
- 2) Split sentences into sub-trees that are phrases for each type: verb, noun, prepositional and others. These sub-trees are overlapping. The sub-trees are coded so that the information about their occurrence in the full tree is retained.
- 3) All the sub-trees are grouped by phrase types.
- 4) Extend the list of phrases by adding equivalence transformations (Section 3.2).
- 5) Generalize each pair of sub-trees for both sentences for each phrase type.
- 6) For each pair of sub-trees, yield an alignment [58], and generalize each node for this alignment. Calculate the score for the obtained set of trees (generalization results).
- 7) For each pair of sub-trees of phrases, select the set of generalizations with the highest score (the least general).
- 8) Form the sets of generalizations for each phrase type whose elements are the sets of generalizations for that type.
- 9) Filter the list of generalization results: for the list of generalizations for each phrase type, exclude more general elements from the lists of generalization for a given pair of phrases.

For a given pair of words, only a single generalization exists; if words are the same in the same form, the result is a node with this word in this form. We refer to the generalization of words occurring in a syntactic tree as a *word node*. If the word forms are different (e.g., one is single and the other is plural), only the lemma of the word remains. If the words are different and only the parts of speech are the same, the resultant node contains only the part-of-speech information with no lemma. If the parts of speech are different, the generalization node is empty.

For a pair of phrases, the generalization includes all the *maximum* ordered sets of generalization nodes for the words in the phrases so that the order of words is retained. Consider the following example:

To buy the digital camera today, on Monday

The digital camera was a good buy today, the first Monday of the month

The generalization is {<*JJ-digital*, *NN-camera*>, <*NN-today*, *ADV,Monday*>}, where the generalization for the noun phrase is followed by the generalization for the adverbial phrase. The verb *buy* is excluded from both generalizations because it occurs in a different order in the above phrases. *Buy - digital - camera* is not a generalization phrase because *buy* occurs in a different sequence in the other generalization nodes.

We can see that the multiple maximum generalizations occur depending on how the correspondence between words is established; multiple generalizations are possible. Ordinarily, the total of the generalizations forms a lattice. To obey the condition of the maximum, we introduce a score for generalization. The scoring weights of generalizations are decreasing, roughly, in the following order: nouns and verbs, other parts of speech, and nodes with no lemma, only a part of speech. In its style, the generalization operation follows the notion of the ‘least-general generalization’ or anti-unification, if a node is a formula in a language of logic. Therefore, we can refer to the syntactic tree generalization as an operation of *anti-unification of syntactic trees*.

To optimize the calculation of the generalization score, we conducted a computational study to determine the POS weights and deliver the most accurate similarity measure possible between sentences [15,57]. The problem was formulated as finding the optimal weights for nouns, adjectives, verbs and their forms (such as gerund and past tense) so that the resultant search relevance is maximized. The search relevance was measured as a deviation in the order of search results from the best result for a given query (delivered by Google). The current search order was determined based on the score of generalization for the given set of POS weights (the other generalization parameters having been fixed). As a result of this optimization, we obtained $W_{NN} = 1.0$, $W_{JJ} = 0.32$, $W_{RB} = 0.71$, $W_{CD} = 0.64$, $W_{VB} = 0.83$, and $W_{PRP} = 0.35$, excluding common and frequent verbs like *get/ take/set/put*, for which $W_{VBcommon} = 0.57$. We also establish that $W_{<POS,*>} = 0.2$ (different words but the same POS), and $W_{<*,word>} = 0.3$ (the same word occurring as different POSs in two sentences).

The generalization score (or the similarity between the sentences $sent_1$ and $sent_2$) can then be expressed as a sum through the phrases of the weighted sum for the words

$$\text{word}_{sent_1} \text{ and } \text{word}_{sent_2} \\ \text{score}(sent_1, sent_2) = \sum_{\{NP, VP, \dots\}} \sum_{POS} W_{POS} \text{word generalization}(\text{word}_{sent_1} \text{ word}_{sent_2}).$$

The (maximal) generalization can then be defined as the generalization with the highest score. Accordingly, we define the generalization for phrases, sentences and paragraphs. The result of the generalization can be further generalized with other parse trees or generalizations. For a set of sentences, the totality of the generalizations forms a lattice; the order of the generalizations is set by the subsumption relation and the generalization score. We enforce the associativity of the generalization of parse trees by means of computation: it must be verified and the resultant list must be extended each time a new sentence is added. Note that such an associativity is not implied in our definition of generalization.

3.2. Nearest neighbor learning of generalizations

To perform the classification, we apply a simple learning approach to the parse tree generalization results. The simplest decision mechanism can be based on maximizing the score of the generalization of an input sentence and a member of the training class. However, to maintain the deterministic flavor of our approach, we select the nearest neighbor method with a limitation for both classes to be classified, as well as foreign classes. The following conditions hold when a sentence U is assigned to a class R^+ and not to the class R^- :

1. U has a nonempty generalization (having a score above the threshold) with a positive example R^+ . It is possible that U also has a nonempty common generalization with a negative example R^- ; its score should be below that of R^+ (meaning that the graph is similar to both the positive and negative examples).
2. For any negative example R^- , if U is similar to R^- (i.e., $U * R^- \neq \emptyset$) then $\text{generalization}(U, R^-)$ should be a sub-tree of $\text{generalization}(U, R^+)$. This condition introduces the partial order to the measure of similarity. It states that to be assigned to a class, the similarity between the current sentence U and the closest (in terms of generalization) sentence from the positive class should be higher than the similarity between U and each negative example (please also compare with [34]).

3.3. Equivalence transformation of phrases

We have manually created and collected a rule base for generic linguistic phenomena from various resources. Unlike a text entailment system, in our situation we do not require a complete transformation system as long as we have a sufficiently rich set of examples. The transformation rules were developed under the assumption that informative sentences should have relatively simple structures [28].

Syntactic-based rules capture the entailment inferences associated with common syntactic structures, including the simplification of the original parse tree, reducing it into a canonical form, extracting its embedded propositions, and inferring propositions from the nonpropositional sub-trees of the source tree (Table 1).

The valid matching of sentence parts embedded as verb complements depends on the verbs' properties and the polarity of the context in which the verb appears (positive, negative, or unknown). We used the list of verbs for communicative actions in Galitsky and Kuznetsov [13], which indicate positive polarity context. This list is complemented with a few reporting verbs, such as *say* and *announce*, because opinions are often provided in reported speech in the news domain, while authors are usually considered reliable. We also used annotation rules to mark the negation and modality of predicates (mainly verbs) based on their descendent modifiers.

Table 1

Rules of graph reduction for generic linguistic structures. The resultant reductions are italicized.

Category	Original/Transformed fragment
conjunctions	Camera is very stable and <i>has played an important role in filming their wedding</i>
clausal modifiers	The flash was disconnected when the <i>children went out to play in the yard</i>
relative clauses	I was forced to close the LCD, which <i>was blinded by the sun</i>
Appositives	<i>Digital zoom</i> , a feature provided by the new generation of cameras, dramatically decreases the images' sharpness
Determiners	My customers use their (an auto ...) autofocus camera for polar expeditions (their => an)
Passive	A cell phone can be easily grasped in the palm of a hand (<i>The hand can easily grasp the cell phone in its palm</i>)
genitive modifier	Sony's LCD screens work as well as Canon's in a sunny environment (<i>The LCD of Sony... as well as that of Canon</i>)
Polarity	It made me use digital zoom for mountain shots (<i>I used digital zoom...</i>)

An important class of transformation rules involves noun phrases. The adjectives of a single noun group can be re-sorted, as well as all nouns except the head one. A noun phrase that is a post-modifier of the head noun of a given phrase can be merged with the latter. The resultant meaning may be distorted; otherwise, we would miss important commonalities between expressions containing noun phrases. For an expression 'NP₁ <of or for> NP₂', we form a single NP with the head noun *head*(NP₂), *head*(NP₁) playing the role of modifier, and an arbitrary sorting of adjectives. For example, we convert 'camera with digital zoom' into 'digital zoom camera', *head*(NP) = camera.

3.4. Simplified example of the generalization of sentences

We present an example of the generalization operations of two sentences. The intermediate sub-trees are shown as lists for brevity. The generalization of distinct values is denoted by **. Let us consider the three following sentences:

I am curious how to use the digital zoom of this camera for filming insects.

How can I get short focus zoom lens for digital camera?

Can I get auto focus lens for digital camera?

We first draw the parse trees for these sentences and determine how to build their maximal common sub-trees (Fig. 1).

We can see that the second and third trees are quite similar. Therefore, it is simple to build their common sub-tree as an (interrupted) path of the tree (Fig. 2):

{MD-can, PRP-I, VB-get, NN-focus, NN-lens, IN-for JJ-digital NN-camera}. At the phrase level, we obtain:

Noun phrases: [[NN-focus NN-*], [JJ-digital NN-camera]]

Verb phrases: [[VB-get NN-focus NN-* NN-lens IN-for JJ-digital NN-camera]]

One can see that common words remain in the maximum common sub-tree, except 'can', which is unique to the second sentence, and the modifiers of 'lens', which are different between the two sentences (shown as *NN-focus NN-* NN-lens*). When sentences are not as similar to one another as sentences 2 and 3 are, we proceed to their generalization on a phrase-by-phrase basis. Below, we express the syntactic parse tree via chunking [56], using the format <position (POS – phrase)>.

Parse 1 0(S-I am curious how to use the digital zoom of this camera for filming insects), 0(NP-I), 2(VP-am curious how to use the digital zoom of this camera for filming insects),

2(VBP-am),

5(ADJP-curious), 5(JJ-curious),

13(SBAR-how to use the digital zoom of this camera for filming insects), 13(WHADVP-how), 13(WRB-how), 17(S-to use the digital zoom of this camera for filming insects),

17(VP-to use the digital zoom of this camera for filming insects), 17(TO-to),

20(VP-use the digital zoom of this camera for filming insects), 20(VB-use),

24(NP-the digital zoom of this camera), 24(NP-the digital zoom), 24(DT-the),

28(JJ-digital),

36(NN-zoom), 41(PP-of this camera), 41(IN-of), 44(NP-this camera), 44(DT-this),

49(NN-camera), 56(PP-for filming insects), 56(IN-for),

60(NP-filming insects), 60(VBG-filming), 68(NNS-insects)

Parse 2

[0(SBARQ-How can I get short focus zoom lens for digital camera), 0(WHADVP-How), 0(WRB-How), 4(SQ-can I get short focus zoom lens for digital camera), 4(MD-can), 8(NP-I), 8(PRP-I), 10(VP-get short focus zoom lens for digital camera), 10(VB-get), 14(NP-short focus zoom lens), 14(JJ-short), 20(NN-focus), 26(NN-zoom), 31(NN-lens),

36(PP-for digital camera), 36(IN-for), 40(NP-digital camera), 40(JJ-digital), 48(NN-camera)]

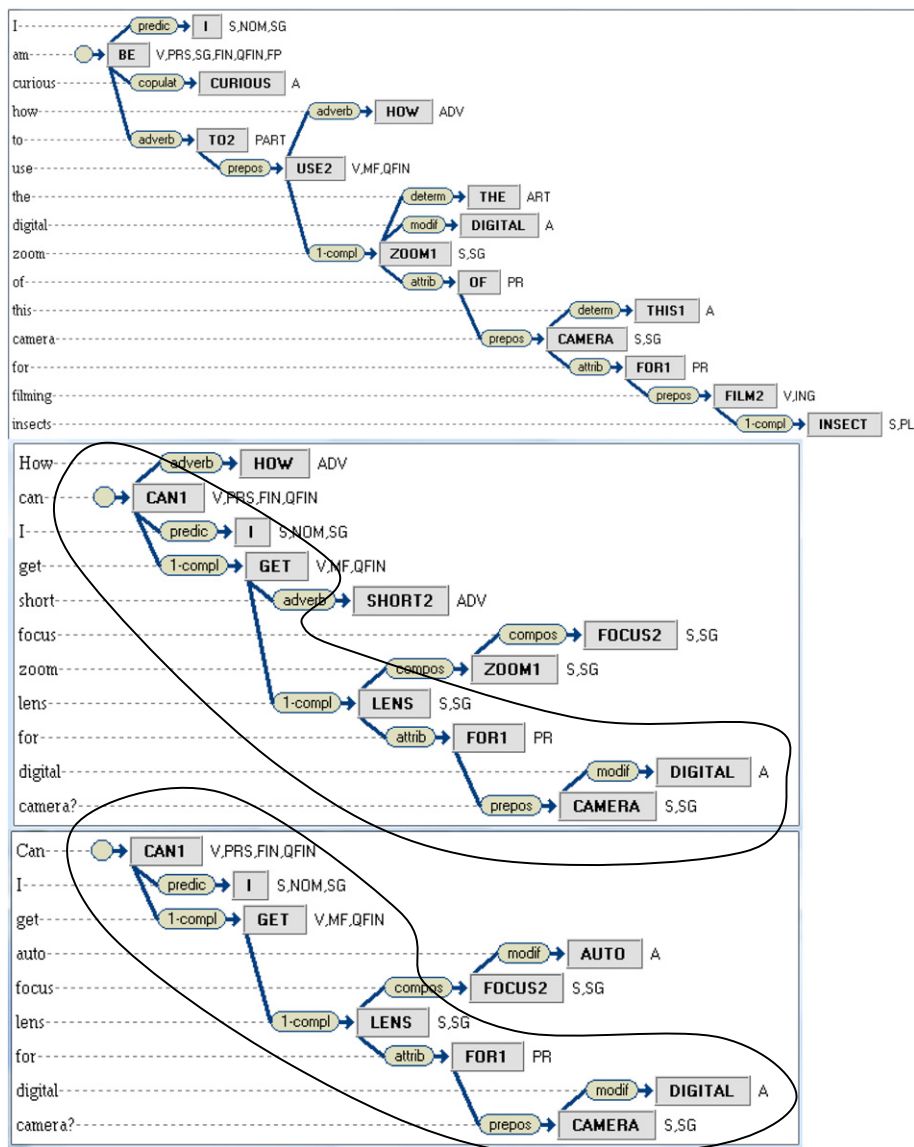


Fig. 1. Parse trees for three sentences. The curve shows the common sub-tree (in this case, there is only one) for the second and third sentences.

We can now group the above phrases by their phrase type [NP, VP, PP, ADJP, WHADVP]. The numbers at the beginning of each phrase encode their character positions. Each group contains the phrases of the same type because the matches occur between the same types.

Grouped phrases 1 [[NP [DT-the JJ-digital NN-zoom IN-of DT-this NN-camera], NP [DT-the JJ-digital NN-zoom], NP [DT-this NN-camera], NP [VBG-filming NNS-insects]], [VP [VBP-am ADJP-curious WHADVP-how TO-to VB-use DT-the JJ-digital NN-zoom IN-of DT-this NN-camera IN-for VBG-filming NNS-insects], VP [TO-to VB-use DT-the JJ-digital NN-zoom IN-of DT-this NN-camera IN-for VBG-filming NNS-insects], VP [VB-use DT-the JJ-digital NN-zoom IN-of DT-this NN-camera IN-for VBG-filming NNS-insects]], [], [PP [IN-of DT-this NN-camera], PP [IN-for VBG-filming NNS-insects]], [], [], []]

Grouped phrases 2 [[NP [JJ-short NN-focus NN-zoom NN-lens], NP [JJ-digital NN-camera]], [VP [VB-get JJ-short NN-focus NN-zoom NN-lens IN-for JJ-digital NN-camera]], [], [PP [IN-for JJ-digital NN-camera]], [], [], [SBARQ [WHADVP-How MD-can NP-I VB-get JJ-short NN-focus NN-zoom NN-lens IN-for JJ-digital NN-camera], SQ [MD-can NP-I VB-get JJ-short NN-focus NN-zoom NN-lens IN-for JJ-digital NN-camera]]]

3.4.1. Sample generalization between phrases

At the phrase level, generalization starts with finding the alignment between two phrases, where we attempt to set correspondences between as many words as possible in the two phrases. We ensure that the alignment operation retains the

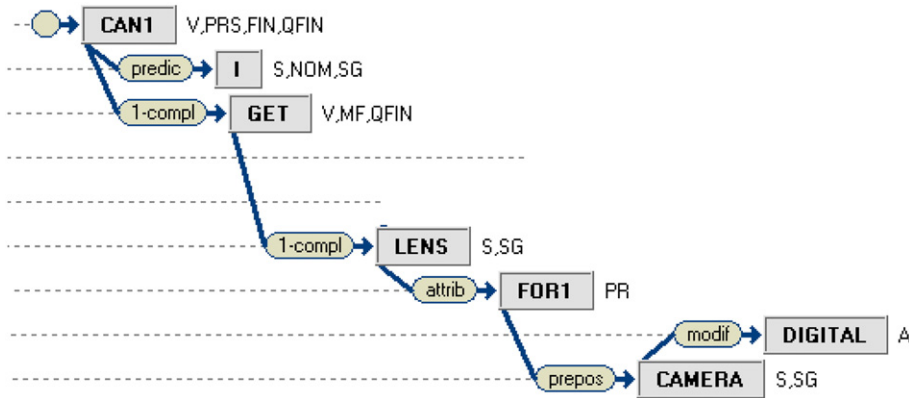


Fig. 2. Generalization results for the second and third sentences.

integrity of the phrase: in particular, two phrases can be aligned only if the correspondence between their head nouns is established. A similar integrity constraint applies to aligning verb, prepositional and other types of phrases.

[VB-use DT-the JJ-digital NN-zoom IN-of DT-this NN-camera IN-for VBG-filming NNS-insects]

∩

[VB-get JJ-short NN-focus NN-zoom NN-lens IN-for JJ-digital NN-camera]

=

[VB-* JJ-* NN-zoom NN-* IN-for NN-*]

Here, we present the mapping between either words or the respective POS to explain how generalization occurs for each pair of phrases of each phrase type. Six mapping links between the phrases correspond to the six members of the generalization results' links. The resultant generalization is shown in bold in the example below for verb phrases (VP).

3.4.2. Generalization result

NP [[JJ-* NN-zoom NN-*], [JJ-digital NN-camera]]

VP [[VBP-* ADJP-* NN-zoom NN-camera], **[VB-* JJ-* NN-zoom NN-* IN-for NN-*]**]

PP [[IN-* NN-camera], [IN-for NN-*]]

score(NP) = ($W_{<POS,*>} + W_{NN} + W_{<POS,*>}$) + ($W_{NN} + W_{NN}$) = 3.4,

score(VP) = ($2 * W_{<POS,*>} + 2 * W_{NN}$) + ($4 W_{<POS,*>} + W_{NN} + W_{PRP}$) = 4.55, and

score(PP) = ($W_{<POS,*>} + W_{NN}$) + ($W_{PRP} + W_{NN}$) = 2.55,

therefore, score = 10.5.

One can see that such a common concept as *digital camera* is automatically generalized from the examples, as well as the verb phrase *be some-kind-of zoom camera*, which expresses a common meaning between the above sentences. Note the occurrence of the expression [digital-camera] in the first sentence: although *digital* does not refer to *camera* directly, when we merge the two noun groups, *digital* becomes one of the adjectives of the resultant noun group with the head *camera*. It is matched against the noun phrase reformulated in a similar way (but with the preposition *for*) in the second sentence with the same head noun *camera*. We present more complex generalization examples in Section 4.

3.5. Generalizing semantic role expressions

In this Section, we demonstrate how generalization can be extended toward the semantic levels to ensure that the types of arguments of the linguistic predicates agree. We intend to avoid successful matches of the same word in two NL expressions when the word has one semantic role in the first expression and *another* role in the second expression. We employ Semantic Role Labeling to introduce an additional match constraint to enforce the agreement of the semantic roles.

Semantic role labeling (SRL) is a series of approaches to low-level semantic representations where for each verb in a sentence, the goal is to identify all the constituents that fill a semantic role and determine their roles, such as Agent, Patient, Instrument, or their adjuncts [7,9,11,19,20,37]. The *semantic role* in language is the relationship between a syntactic constituent and a linguistic predicate. Recognizing and labeling semantic arguments is a key task for answering “Who,” “When,” “What,” “Where,” “Why” questions in information extraction, question answering, and summarization. SRL results in the following types of arguments:

A0 ... A5 arguments associated with a verb predicate, as defined in the PropBank Frames scheme [43]. In a typical case, A0 is whoever is performing an action, A1 is the subject, and A2 is the purpose.

AM- <i>T</i>	adjunctive arguments of various sorts, where <i>T</i> is the type of the adjunct. These types include locative, temporal (TMP), manner, and modal (MOD).
AA	causative agents.
V	the verb of the proposition.
R-*	a reference to some other argument of A* type.

Let us produce a generalization of LSR on two sentences from the example in Section 3.3. In the second row, semantic roles are subscribed for the verbs: for the verb *use* we have A0 – *I*, A1 – *zoom of this camera* and A2 – *filming insects*. For the verb *filming* we only have A1 – *insects*.

I am curious how to use the digital zoom of this camera for filming insects.

(A0) use (A1) (A2)

film

(A1)

How can I get short focus zoom lens for digital camera.

(A0) get (A1)

The matching result when semantic roles are taken into account is [*I*-A0] verb [*zoom*-A1].

One can see that SRL can serve as an additional constraint on syntactic generalization, providing one more step toward semantic generalization. This follows along the line of Moreda et al. [25,46] who proposed using corpus-based semantic role information as an extension of an Information Retrieval system in order to reduce the number of documents or passages retrieved by the system.

3.6. Other extensions of anti-unification

Because we describe a version of anti-unification applied to linguistic structures, it is worth discussing other cases of anti-unification. In addition to the generalization of literals, Plotkin [26] describes an algorithm for θ – the least generalization of clauses. A clause C_1 generalizes a clause C_2 (denoted by $C_1 \leq C_2$) if C_1 subsumes C_2 , i.e., a substitution θ exists such that $C_1\theta \leq C_2$; this type of substitution is called subsumption [3]. A generalization C_1 of a clause C_2 can be obtained by applying a θ -subsumption-based generalization operator Θ that maps a clause C_2 to a set of clauses $\Theta(C_2)$ (generalizations of C_2). Informally speaking, if clause C θ -subsumes clause D , then D can be converted to C by (1) dropping premises and (2) turning constants into variables. A clause C is a least generalization of a set of clauses S if

1. C generalizes each clause in S : $\forall E \in S \leq C$;
2. C is the smallest clause satisfying condition 1: $\exists D \forall E \in S D \leq E \Rightarrow D \leq C$.

Now let us consider a dynamic set of clauses S . Referring to implication instead of the weaker subsumption relationship presented above would also consider the generalization with respect to the current knowledge. A generalization relative to a set of clauses P is defined as follows: A clause C generalizes a clause D relative to a set of clauses P if a substitution θ exists such that $P \models C\theta \rightarrow D$. The *generalized subsumption* of definite Horn clauses as an extension of θ -subsumption is defined with the restriction that the corresponding clause heads must refer to the same concept. Informally speaking, if a clause C generally subsumes clause D , then C can be converted to D by (1) turning variables into constants or other terms, (2) adding atoms to the body, and (3) partially evaluating the body by resolving some clause in P with an atom in the body. The third conversion process is additional to the conversion for θ -subsumption.

The subsumption relationship plays an important role in reducing the list of the anti-unification of syntactic trees as well. If a parse tree for a phrase C subsumes the parse tree for a phrase D , then D can be converted to C via the operations described in Section (3.2) and reductions like removing the determiners, adjectives, complements and modifiers from the noun phrases.

3.7. From syntax to inductive semantics

To demonstrate how syntactic generalization allows us to ascend from the syntactic to the semantic level, we follow Mill's *direct method of agreement* (induction) as applied to linguistic structures. As the British philosopher JS Mills wrote in his 1843 book 'A System of Logic', "If two or more instances of the phenomenon under investigation have only one circumstance in common, the circumstance in which alone all the instances agree, is the cause (or effect) of the given phenomenon [64]."

Consider a linguistic property A of a phrase f . For A to be a necessary condition of an effect E , A must always be present in multiple phrases that deal with E . Therefore, we determine whether the linguistic properties considered the 'possible necessary conditions' are present or absent in the sentence. Obviously, any linguistic properties that are absent when a meaning is present cannot be necessary conditions to that meaning of a phrase.

For example, the method of agreement can be represented as a phrase f_1 , where the words {A B C D} occur together with the meaning formally expressed as $\langle w x y z \rangle$. In addition, consider another phrase f_2 , where the words {A E F G} occur together with the same meaning $\langle w t u v \rangle$ in phrase f_1 . By applying generalization to the words {A B C D} and {A E F G}, we obtain {A} (here, for the sake of the example, we ignore the syntactic structures of f_1 and f_2). Therefore, we can see that word A is the cause of w (has meaning w).

In this way, we can produce (inductive) semantics by applying syntactic generalization. *Semantics cannot be obtained given only syntactic information; however, by generalizing two or more phrases, we obtain an (inductive) semantic structure.*

4. From generalization to logical form representation

We now demonstrate how the generalization framework can be combined with semantic representations, such as logic forms, to perform the learning of a text's meaning. We have demonstrated how semantic features can be deduced from syntactic parse trees when an appropriate similarity operation is found. However, in a number of applications, certain semantic knowledge is available in advance and therefore, does not have to be learned. In this section, we show how to combine preset semantic information with learned information to build the most accurate semantic representation.

We use notes on a number of customers of a bank. The dataset of five paragraphs is introduced and then illustrated as a step-by-step learning procedure.

- 1p. A friend transferred funds from a checking to a savings account. He then used the savings funds to pay for his mortgage.
 2p. Premier account customers decided to transfer their funds from premier to regular savings accounts. The couple then used their premier account for automated mortgage payments.
 3p. A mortgage customer transferred the mortgage account from fixed to adjustable. She then decided to use the remaining funds as the last mortgage payment for her second home.
 1n. A broker transferred his title from a corporate brokerage to individual accounts. He used it to deposit significant personal funds into the brokerage account.
 2n. A manager transferred rents from rent collection corporate to investment accounts. In the past, he used to deposit rents in his investment brokerage account directly.

To demonstrate a deep level of understanding of the meanings of these short paragraphs, let us introduce the two classes of “individual bank users” and “corporate bank users” and demonstrate how these classes can be formed from our data and the performed classification. Note that there is no explicit indicator of belonging to one of this classes; it must be inferred from the text. There could be other classes in which the semantic information has to be inferred, such as ‘obtained funds are used for something’ and ‘no such statement is made’, ‘account type transfer’ and ‘refinancing’, and many more. We use the denotation {1p, 2p, 3p} for the set of positive examples and {1n, 2n} for the set of negative examples. For each example, we enumerate the sentences in the paragraphs as {a, b, ...}.

We intend to express the commonalities between the elements of training set to ‘explain’ membership in a class, following the classical methodology of induction [23]. We hypothesize that the common linguistic features of a training set *cause* the target feature (the class). In this section, we form these features on both syntactic levels by means of generalization and on the semantic level by means of logical anti-unification. For this purpose, we will first proceed on the syntactic level and then demonstrate its performance on the semantic level of logic forms. Finally, we will show how the syntactic level generalizations can be mapped onto the semantic level generalizations.

We build a lattice of generalizations separately for the positive and negative sets of the paragraphs. The order of generalization is selected so that it results in the maximal lattice, in the sense that the total score of all the expressions of the nodes is the highest possible. For example, for three paragraphs, it always produces a higher score to generalize the two paragraphs with a high similarity to each other first. The machine learning procedure (such as Nearest Neighbor) will then relate a new paragraph to a lattice of training set examples of either class. The learning procedure occurs on both the syntactic and semantic levels as well, as both syntactic and semantic properties of the text can be the criteria of belonging to a class.

The lattice built for three paragraphs of the positive set is shown in Fig. 3a. For each lemma in the generalization result, we use a simplified denotation: either the lemma itself (if available) or the POS (if the lemmas are different). Single lines depict the generalizations for the first sentence of each paragraph, (a), and double lines represent the second sentence, (b). There are multiple sentences appearing in different orders in a more general case. The lattice depicts the relations of being “more general” between the generalization results.

4.1. Mapping into logic forms

We intend to explore the inter-relations between the generalization operation on sentences and the anti-unification operation on the logical representations of these sentences. We select a small domain and attempt to classify sentences, applying both the former and latter operations to observe how we can operate at the syntactic and semantic levels.

To define the mapping into logic forms, we must form logical predicates and specify the semantic types of their arguments. We do not believe that semantic types can be adequately defined at the time of text processing (automatically): if this were the case, we would need to continue adding new semantic types as we encountered new co-occurrences of words for predicates and instances for their selected arguments. For the selected domain (financial services), we form the logical predicates listed below to represent the meanings of the entities in our sample paragraphs. These definitions follow a semantic role labeling style (as in 'transfer') with additional domain-specific constraints on the arguments. Additionally, nouns can form logical predicates if they express entities important to the current domain, such as *account*. We use square brackets for comments.

transfer(who [agent], what [from-what], to-what [result]).

use(who, what [e.g. funds], for-what [for certain purpose]).

account(type [standard account type like checking/saving], attribute [all other account parameters together]).

deposit(who, what [which funds], to-what[account]).

All other logical predicates in this domain have only a single argument for each attribute: *mortgage*(attribute), *customer*(attribute), *rent*(attribute[action with rent]), *title*(attribute[what kind of title]).

Unlike the generalization operation, here, synonyms are used to build logic forms. Synonyms are very domain-specific: for example, *account* = *fund* in expression [VBX-deposit PRP-to]. In other cases (domains), such as 'software user accounts', *account* and *fund* cannot be synonyms.

The only 'black box' part of our current consideration is the procedure of building logic forms from texts or generalization results. This rule-based procedure starts by identifying predicates from the lemmas in the sentences and identifying the words

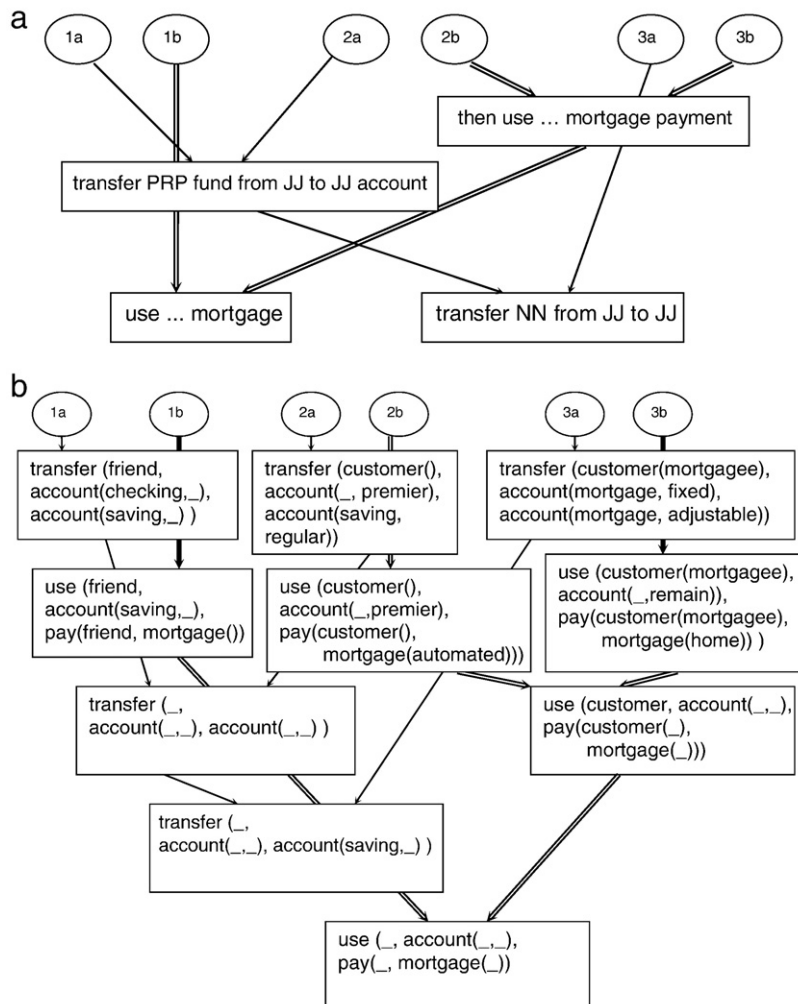


Fig. 3. a: The lattice of generalizations for three paragraphs from positive set. b: The lattice of anti-unifications for three paragraphs. c: Mapping between syntactic and semantic levels. Results of the generalization of sentences are mapped into the anti-unification of the respective logic forms. d: Lattice of syntactic commonalities for the negative set. e: Lattice of anti-unifications for the negative set.

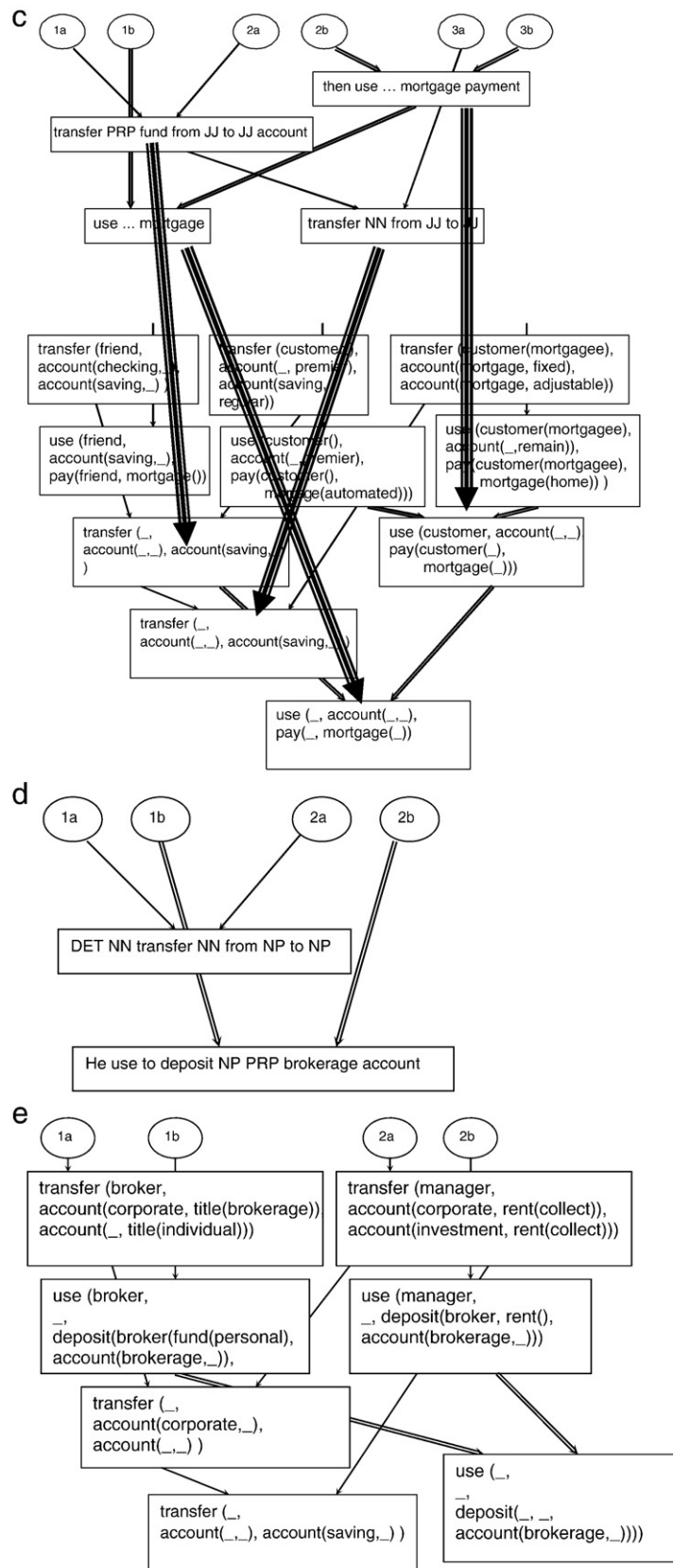


Fig. 3 (continued).

that are syntactically linked to them (as determined by the syntactic tree) to be substituted into the proper arguments of the predicates. The next step is to link the remaining uninstantiated variables of the predicates together, assuming they are the same objects and again applying rules to the parse tree. The procedure of building logic forms from text is thoroughly presented elsewhere [1], including a book by Galitsky [12]. Recently, statistical approaches have attempted to build logic forms in the form of unsupervised learning without building rules by hand [47].

Commonalities between paragraphs of positive classes at the semantic level are depicted in Fig. 3b. A lattice shows the order (in terms of generality) between the logic forms for original sentences and paragraphs and between the anti-unifications results for the logic forms. The anonymous variables ‘_’ are used to indicate the predicates’ arguments that are either

- 1) not instantiated, or
- 2) obtained as the results of anti-unifications in which the values are different.

Analogously to the syntactic level, single lines correspond to the first sentence and double lines to the second sentence.

We depict the mapping between the syntactic level of Fig. 3a and the semantic level of Fig. 3b in Fig. 3c. We can first generalize each pair of sentences and then translate the results into a logic form. Alternatively, we can translate each sentence into a logic form first and then anti-unify those logic forms. Obviously, in simple cases, the results are identical; however, in general, this does not hold true. Both operations lead to the loss of information of various sorts.

Fig. 3c shows multiple paths to the results of the operations of generalization and anti-unification. There is a criterion for the optimal path: the resultant score of expression. For a logic form, the score is the number of terms in the expression; this fits the score of generalization well. We define the *optimal path* to the logic form of a set of samples as the one which leads to the resultant logic form with the *highest score*. Exhaustive iteration through the paths is used to obtain such logic form.

The optimality of the paths for finding syntactic and semantic commonalities between text paragraphs is grounded in linguistic features. For example, if anti-unification precedes generalization, using semantic operations such as synonym substitution and anaphora resolution makes the resultant expressions more complete. If we build a logic form from two sentences:

predicate1(customer, ...) [from the first sentence] and

predicate2(he, ...) [from the second sentence]—we can apply the obtained fact that ‘he’ = ‘customer’ into resultant form. Hence, anaphora resolution does not always commute with the generalization operation.

predicate1(customer,...) & predicate2(customer,...). Otherwise, if we apply the generalization first, we will not be able to apply the anaphora resolution and the resultant logic form will miss the value ‘customer’. The reader can see that when the generalization results are mapped into ‘richer’ logic form representation, one of the above semantic operations occurs.

At the semantic level, the ‘longer’ the natural language expression to be represented is, the more information can be represented by a logic form. On the contrary, for generalization, the more complex the natural language expression is, the more likely equivalent transformation rules (Section 3.2) for matching will run into prohibitive conditions.

The reader may observe that attempting to perform the presented learning scenario using the *bag of words* approach instead of the syntactic and semantic levels presented in this section will most likely fail because more specific linguistic signals have to be handled: particularly, the specific semantic of *used-to* associated with the negative class. Both the positive and negative classes include the keyword ‘used’, but learning the structural difference (the second argument is not substituted) and, therefore, the semantic difference helps to explicitly formulate the criteria for the classes.

5. Syntactic generalization-based search engine and its evaluation

The search engine based on syntactic generalization is designed to provide opinion data in an aggregated form obtained from various sources. Conventional search results and Google-sponsored link formats are selected because they are the most effective and are already accepted by a vast community of users.

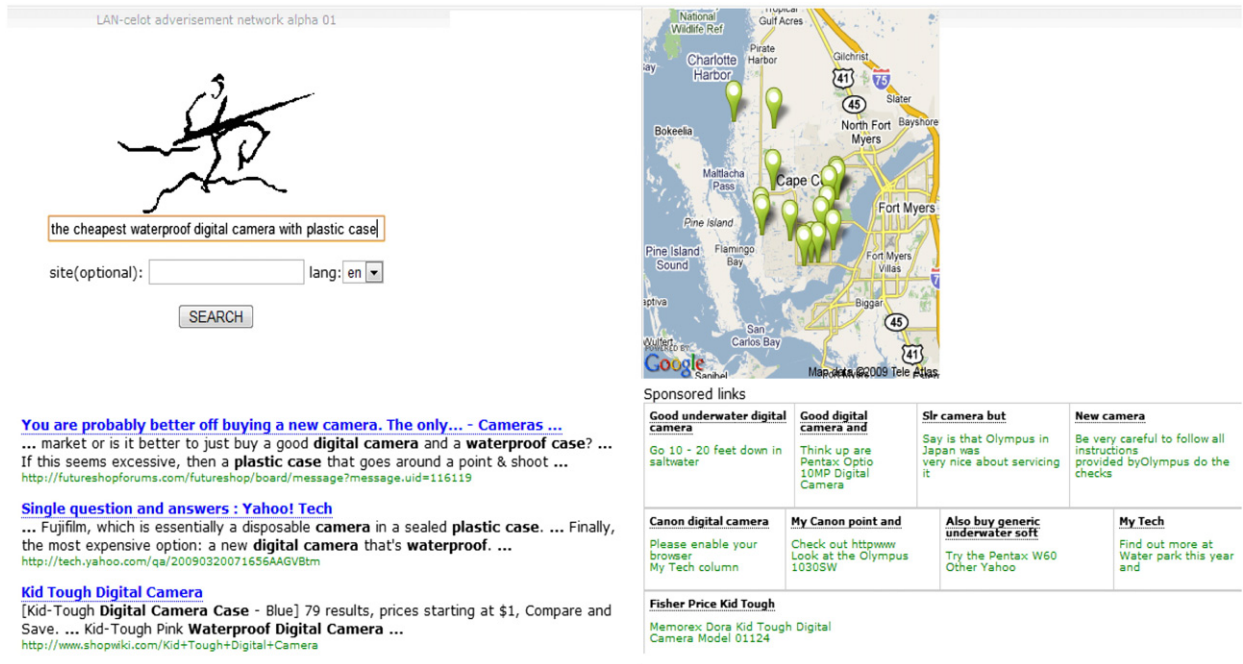
5.1. User interface of search engine

The user interface is shown in Fig. 4. To search for an opinion, a user specifies a product class, a name of particular products, and a set of its features, specific concerns, needs or interests. A search can be narrowed down to a particular source; otherwise, multiple sources of opinion (review portals, vendor-owned reviews, forums and blogs available for indexing) are combined.

The opinion search results are shown on the bottom left. For each result, a snapshot is generated indicating a product, its features that the system attempts to match to a user opinion request, and sentiments. In case of multiple sentence queries, a hit contains a combined snapshot of multiple opinions from multiple sources, dynamically linked to match the user request.

Automatically generated product ads compliant with the Google-sponsored link format are shown on the right. The phrases in the generated ads are extracted from the original products’ web pages and may be modified for compatibility, compactness and their appeal to potential users. There is a one-to-one correspondence between the products in the opinion hits on the left and the generated ads on the right (unlike in Google, where the sponsored links list different websites from those presented on the left).

LAN-celot advertisement network alpha 01



the cheapest waterproof digital camera with plastic case

site(optional): lang:

SEARCH

You are probably better off buying a new camera. The only... - Cameras ...
 ... market or is it better to just buy a good **digital camera** and a **waterproof case**? ...
 If this seems excessive, then a **plastic case** that goes around a point & shoot ...
<http://futureshopforums.com/futureshop/board/message?message.uid=116119>

Single question and answers : Yahoo! Tech
 ... Fujifilm, which is essentially a disposable **camera** in a sealed **plastic case**. ... Finally,
 the most expensive option: a new **digital camera** that's **waterproof**. ...
<http://tech.yahoo.com/qa/20090320071656AAGVBtm>

Kid Tough Digital Camera
 [Kid-Tough **Digital Camera Case - Blue**] 79 results, prices starting at \$1, Compare and
 Save. ... Kid-Tough Pink **Waterproof Digital Camera** ...
<http://www.shopwiki.com/Kid+Tough+Digital+Camera>

Sponsored links

Good underwater digital camera Go 10 - 20 feet down in saltwater	Good digital camera and Think up are Pentax Optio 10MP Digital Camera	Slr camera but Say is that Olympus in Japan was very nice about servicing it	New camera Be very careful to follow all instructions provided by Olympus do the checks
Canon digital camera Please enable your browser My Tech column	My Canon point and Check out httpwww Look at the Olympus 1030SW	Also buy generic underwater soft Try the Pentax W60 Other Yahoo	My Tech Find out more at Water park this year and
Fisher Price Kid Tough Memorex Dora Kid Tough Digital Camera Model 01124			

Fig. 4. User interface of generalization-based search engine.

Both respective business representatives and product users are encouraged to edit and add ads, expressing product feature highlights and usability opinions, respectively. This feature assures openness and community participation in providing access to linked opinions for other users. For example, a *negative* experience staying in a hotel can be expressed as follows:

Disney World Super 12 Motel
Safest place in the area
Take your car there and have it stolen

A search phrase may combine multiple sentences: for example: “*I am a beginning user of digital cameras. I want to take pictures of my kids and pets. Sometimes I take it outdoors, so it should be waterproof to resist the rain*”. Obviously, this type of specific opinion request can hardly be represented by keywords like ‘beginner digital camera kids pets waterproof rain’. For a multi-sentence query, the results are provided as linked search hits:

Take Pictures of Your Kids? ... Canon 400D EOS Rebel XTi **digital SLR camera** review ↔ I am by no means a professional or long-time user of SLR cameras.
How To Take Pictures Of Pets And Kids ... Need help with **Digital slr camera** please!!!! - Yahoo! Answers ↔ I am a **beginner** in the world of the **digital SLR** ...
 Canon 400D EOS Rebel XTi **digital SLR camera** review (Website Design Tips) / Animal, **pet, children**, equine, livestock, farm portrait and stock ↔ I am a **beginner** to the slr **camera** world. ↔ I want to **take** the best **picture** possible because I know you. Call anytime.

Linking (↔) is determined in real time to address each part of a multi-sentence query, which may be a blog posting seeking advice. Linked search results provide comprehensive opinions on the topic of the user's interest, obtained from various sources and linked on the fly [17].

5.2. Qualitative evaluation of search

Obviously, the generalization-based search performance is higher for longer keyword queries and natural language queries, in which a high-sensitivity comparison of query and search results allows a determination of the semantic relevancy between them.

*We start with the example query “National Museum of Art in New York” (Fig. 5), which illustrates a typical search situation in which the user does not know the exact name of an entity. We present the results as ordered by the generalization-based search

NATIONAL MUSEUM OF CATHOLIC ART & HISTORY - New York, NY (#5)
 NATIONAL MUSEUM OF CATHOLIC ART & HISTORY - in New York, NY. Get contact info, directions and more at YELLOWPAGES.COM
 National Academy Museum & School of Fine Arts(#18)
 He is currently represented by Ameringer Yohe Fine Art in New York. ... © 2007 National Academy Museum & School of Fine Arts, New York. Join Our Mailing List ...
 International Council of Museums: Art Galleries(#29)
 (In French and English.) **National Museum of Modern Art.** Musée du ... Metropolitan **Museum of Art, New York City.** One of the largest **art museums** in the world. ...
 Virtual NYC Tour: New York City Museums(#23)
National Museum of the American Indian (New York branch) ... Cloisters is one of the **museums** of the Metropolitan **Museum of Art in New York City.** ...
 Great Museums - SnagFilms(#9)
 Founded in 1870, the Metropolitan **Museum of Art in New York City** is a three ... Home Base: The **National** Baseball Hall of Fame and **Museum** ...
 National Contemporary Art Museum Gets Seoul Venue(#2)
 ... nearby example is the **National Museum of Art in** Deoksu Palace," said ... can also refer to the MoMA's (**Museum of Modern Art**) annex **PSI in New York**," he said. ...
 National Lighthouse Museum New York City.com : Arts ...(#1)
 NYC.com information, maps, directions and reviews
 on **National Lighthouse Museum** and other **Museums in New York City.** NYC.com, the authentic city site, also offer a ...
 National Academy Museum New York City.com : Arts ...(#0)
 NYC.com information, maps, directions and reviews
 on **National Academy Museum** and other **Museums in New York City.** NYC.com, the authentic city site, also offer a ...

Fig. 5. Sample search results for generalization-based search engine.

engine, retaining the information from the original order obtained for this query on Yahoo.com (#x). Note that the expected name of the museum is either *Metropolitan Museum of Art* or *National Museum of Catholic Art & History*.

The matching procedure must verify that 'National' and 'Art' from the query belong to the noun group of the main entity (museum) and that this entity is linguistically connected to 'New York'. If these two conditions are satisfied, we get the first few relevant hits (although mutually inconsistent, they contain either *museum* or *academy*). In Yahoo sorting, we can see that first few relevant hits are numbered #5, #18, and #29. Yahoo's #0 and #1 are at the far bottom of the generalization-based search engine: the above conditions of 'National' and 'Art' are not satisfied, so these hits do not appear to be as relevant. Obviously, conventional search engines would have no problem delivering answers when an entity is named exactly (Google does a good job answering the above query; this is possibly achieved by learning what other people ended up clicking through).

Hence, we observe that generalization helps in queries where the important components and the linguistic link between them in a query should be retained in the relevant answer abstracts. A conventional search engine uses a high number of relevancy dimensions, such as page rank. However, to answer more complex questions, syntactic similarity expressed via generalization presents substantial benefits.

5.3. Anti-unification distance for search relevance

Anti-unification is originally described for trees and can be extended in a straightforward way to directed acyclic graphs. The anti-unifier tree of two trees T_1 and T_2 is obtained by replacing some sub-trees in T_1 and T_2 with special nodes containing term placeholders marked with integers. These nodes can be represented as indexed stars $*_n$. For example, anti-unifier of the terms

$camera(zoom(digital), increase(focus(distance)))$ and
 $camera(zoom(optical), increase(battery(life)))$
 will be $camera(zoom(*_d), increase(*_i))$.

In some abstract syntax tree representations, occurrences of the same variable refer to the same leaf in a tree. The anti-unifier of two trees represents their "skeleton", inserting placeholders for the sub-trees that differ. The anti-unifier of a set of trees can be viewed as the most specific pattern that matches each tree in the set. Therefore, it can be used to store the average value of a set of trees. An anti-unifier stores only the common top-level tree structure. Let us define the anti-unification distance between two trees, following [4,5].

Let U be the anti-unifier of two trees T_1 and T_2 with substitutions σ_1 and σ_2 . Let n be the number of placeholders in U . Then σ_1 and σ_2 are mappings from the set $*_1 \dots *_n$ to the substituting trees. The size of a tree is defined as its number of leaves. This notion of size is robust to the particularities of representing abstract syntax trees because it is equal to the number of all names and

constant occurrences in the program. The anti-unification distance between T_1 and T_2 is defined as the sum of the sizes of all the substituting trees in σ_1 and σ_2 . The anti-unification distance for the above example is $\sigma_1 + \sigma_2 = 2$ ($|\sigma_1| = 1$, $|\sigma_2| = 1$). The anti-unification distance can be viewed as the tree-editing distance [4] with a restricted set of operations. It catches the structural differences between two trees and does not allow the permutation of siblings or changes to the number of child nodes.

5.4. Evaluation of search relevance improvement

The evaluation of the search included an assessment of the classification accuracy for the search results as relevant and irrelevant. Because we used the generalization score between the query and each hit snapshot, we drew a threshold of the five highest score results as the relevant class and the rest of the search results as irrelevant. We used the Yahoo search API and the Bing search API, applying the generalization score to find the highest-scoring hits from first fifty Yahoo and Bing search results (Table 2). We then considered the first five hits with the highest generalization scores (not Yahoo/Bing score) to belong to the class of relevant answers. The third and second rows from the bottom contain the classification results for queries of 3–4 keywords, which is slightly more complex than an average query (3 keywords) and significantly more complex queries of 5–7 keywords, respectively. We used a set of tax-related questions including a tax topic and also some indication of taxpayer's circumstances [12], such as *How can I deduct my medical expense if I became self-employed?*. The evaluation was conducted by the authors.

For a typical search query containing 3–4 words, syntactic generalization is not used. One can see that for a 5–7 word phrases, syntactic generalization deteriorates the accuracy and should not be used. However, for longer queries, the results are encouraging (almost 4% improvement), showing a visible improvement over the current Yahoo and Bing searches once the results are re-sorted based on syntactic generalization. A substantial improvement can be observed in the multi-sentence queries as well.

5.5. Comparison with other means of search relevance improvement

Syntactic generalization was deployed and evaluated in the framework of a Unique European Citizens' attention service (iSAC6+) project, an EU initiative to build a recommendation search engine in a vertical domain. As a part of this initiative, a taxonomy was built to improve the search relevance ([15,57], see also [61]).

This taxonomy is used by matching both the question and answer to a taxonomy tree and relying on the cardinality of the set of overlapping query terms. The comparison of the taxonomy-based score, generalization-based score and hybrid system score is valuable because features of various natures are leveraged (pragmatic, syntactic/semantic and hybrid, respectively).

We built a tool to perform the comparison between the contributions of the above scoring systems (easy4.udg.edu/isac/eng/index.php, de la Rosa, 2007; Lopes Arjona, 2010). The taxonomy learning of the tax domain was conducted in English and then translated into Spanish, French, German and Italian. It was evaluated by project partners using the tool in Fig. 6, where to improve the search precision, a project partner at a particular location modifies the automatically learned taxonomy to fix a particular case, uploads the taxonomy version adjusted for a particular location and verifies the improvement in the relevance. An evaluator can sort by the original Yahoo score, the syntactic generalization score, and the taxonomy score to get a sense of how each of these scores work and how they correlate with the best order of answers for the best relevance.

6. Evaluation of text classification problems

6.1. Comparative performance analysis in text classification domains

To evaluate the expressiveness and sensitivity of the syntactic generalization operation and its associated scoring system, we applied the Nearest Neighbor algorithm to the series of text classification tasks outlined in Section 2 (Table 3). We formed several datasets for each problem, conducted independent evaluation for this dataset and averaged the resultant accuracy (F -measure). The training and evaluation datasets of the texts and the class assignments were made by the authors. Half of each set was used for training and the other half was used for evaluation; the split was random, but no cross validation was conducted. Due to the

Table 2

Evaluation of search relevance improvement using syntactic generalization.

Type of search query	Relevancy of Yahoo search, %, averaging over 10 searches	Relevancy of re-sorting by generalization, %, averaging over 10 searches	Relevancy comp to baseline, %
3–4 Word phrases	77	77	100.0%
5–7 Word phrases	79	78	98.7%
8–10 Word single sentences	77	80	103.9%
2 Sentences, > 8 words total	77	83	107.8%
3 Sentences, > 12 words total	75	82	109.3%

Can Form 1040 EZ be used to claim the earned income credit

You can change the ordering of the table by clicking on column-headers.

[First result](#) [Previous result](#) [Next result](#) [Last result](#)

ORIGINAL-RANK ▾	SYNTACTIC-MATCH SCORE	TAXONOMY-SCORE	TITLE & ABSTRACT
16	3.3	1	2010 Form W-5 Use Form W-5 if you are eligible to get part of t
3	3.3	4	Earned Income Credit Can Form 1040EZ be used to claim the earned i
2	3.3	4	Can Form 1040EZ be used to claim the earned i Can Form 1040EZ be used to claim the earned i
0	3.3	4	Other EITC Issues Question: Can Form 1040EZ be used to claim th
20	3.0	0	Line by Line Tips for Form 1040-EZ (Year 2008) Prepare your 2008 tax returns on Form 1040-EZ
5	3.0	0	Line by Line Tips for Form 1040-EZ (Year 2009) Prepare your 2009 tax returns on Form 1040-EZ
17	2.9	1	FREE 1040EZ - FREE Federal 1040EZ - Federal 10 Now, as an individual, you may wonder whether y
27	2.8	0	2007 Form W-5 I expect to have a qualifying child and be able to
19	2.8	1	2008 Form W-5 I expect to have a qualifying child and be able to

Fig. 6. Sorting search results by taxonomy-based and syntactic generalization scores for a given query “Can Form 1040 EZ be used to claim the earned income credit?”

nature of the problem, positive sets are larger than negative sets for the ad line problems. For the epistemic state classification, the negative set includes all the other epistemic states or no state at all.

We classify each sentence in the corpus using two approaches:

- A baseline WEKA C4.5 as a popular text classification approach
- Syntactic generalization-based approach

We demonstrate that a traditional text-classification approach handles such a complex classification task poorly, due in particular to the slight differences between phrasings in these classes. Using syntactic generalization instead of WEKA C4.5 resulted in a 16.1% increase in the F-measure for the set of digital camera reviews. In the other domains in Table 3, which are more traditional subjects for text classification, we do not expect as dramatic an improvement (not shown).

The lower half of the table contains classification data for the web pages for different products, and the variability in the accuracies can be explained by the various levels of diversity in their phrasings. For example, the ways people express their feelings about cars is much more diverse than their opinions about kitchen appliances. Therefore, the accuracy of the former task is lower than that of the latter. We can see that it is difficult to form verbalized rules for the classes, and the hypotheses are mainly domain-dependent; therefore, substantial coverage of the varieties of phrasing is required.

To form the training set for ad line information extraction, we collected positive examples from existing Google ads, scraping more than 2000 ad lines. The precision of extraction of such lines for the same five categories of products is higher than that of the

Table 3

Accuracies of text classification problems.

Problem domain	Dataset	Data set size (#pos/#neg in each of two classes)	Precision relating to a class, %	Recall relating to a class, %	F-measure
Good for ad line / inappropriate for ad line	digital camera webpages	2000/1000	88.4%	65.6%	75.3%
	wireless services webpages	2000/1000	82.6%	63.1%	71.6%
	laptop webpages	2000/1000	69.2%	64.7%	66.9%
	auto sales webpages	2000/1000	78.5%	63.3%	70.1%
	kitchen appliances webpages	2000/1000	78.0%	68.7%	73.1%
<i>Averages for appropriateness for ad line recognition</i>					
Epistemic state:	Beginner	30/200	77.8%	65.1%	71.4%
	User with average experience	44/200	77.8%	83.5%	80.6%
	Pro or semipro user	25/200	76.2%	81.1%	78.6%
	Potential buyer	60/200	78.7%	84.9%	81.7%
	Open-minded buyer	55/200	73.8%	83.1%	78.2%
	User with one brand in mind	38/200	71.8%	79.6%	75.5%
<i>Averages for epistemic state recognition</i>					
			74.4%	81.9%	78.0%
			75.5%	82.4%	78.7%

a

Contributor Report [Back to report view](#)

News Stories: 21 | Blog Posts: 6 | Videos: 7 | Images: 19 | Comments: 11 | Score Metrics

IMAGES RELATED TO:

Not again! More dead birds fall from sky in Louisiana

by [BorderExplorer](#)

New Roads : LA : USA | about 3 hours ago

[News video footage of Louisiana dead birds incident at top of post] An estimated 500 dead birds were discovered littering a quarter-mile stretch of highway in Point Coupee Parish...

SHARE: [Facebook](#) [Twitter](#) [LinkedIn](#) [StumbleUpon](#) [Reddit](#) [Dribbble](#) [Delicious](#) [Diigo](#) [VK](#) [Odnoklassniki](#) [LiveJournal](#) [MySpace](#) [Flickr](#) [YouTube](#) [Vimeo](#) [SoundCloud](#) [Podcast](#) [RSS](#) [Email](#) [Print](#) [10 retweet](#)

Like [Be the first of your friends to like this.](#)

[Read full report](#)

Reach [Credibility](#) [\[Progress Bar\]](#)

READ MORE: dead birds falling from sky, dead birds, Birds, New Roads, Beebe, Arkansas, environment, technology-news, Labarre, laboratory tests, Louisiana

MORE NEWS FROM: NEW ROADS : LA : USA

CONTRIBUTOR, PARTNER & FEATURED IMAGES [UPLOAD IMAGES](#)



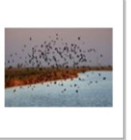


Photo of bird that fell from sky
POSTED BY: [BorderExplorer](#)
Relevance Verifier Results Unavailable
Report Image



As many as 5,000 birds began falling over the ...
IMAGE SOURCE: AFP
Relevance Verifier Results PASSED
Set as report image




Some 500 birds were found dead in Louisiana
IMAGE SOURCE: AFP
Relevance Verifier Results PASSED
Set as report image



One of thousands of blackbirds that fell out of ...
IMAGE SOURCE: Reuters
Relevance Verifier Results PASSED
Set as report image







b

ALLVOICES LOCAL TO GLOBAL NEWS [Help](#) | [Login](#) | [Join](#) [Connect](#) [REPORT YOUR NEWS](#)

Start reporting, reach millions and make money!
[Sign up now](#) on Allvoices

AAA Members get a FREE day from Hertz. Get the first day FREE on a 3-day weekend rental, plus your AAA discount.

All | [Politics](#) | [Sports](#) | [Entertainment](#) | [Business](#) | [Science & Technology](#) | [Conflict & Tragedy](#) | [Odd](#) | [Your Story](#) | [Health](#) | [More](#) [Location or Keyword...](#) [Search](#)

Contributor Report [Back to report view](#)

News Stories: 10 | Blog Posts: 0 | Videos: 7 | Images: 16 | Comments: 2

RELATED NEWS STORIES [ADD A NEWS STORY](#) | [FILTER BY: ALL](#) | [LOCAL](#)

After late intrigue, Walmart bill narrowly advances
San Diego Union-Tribune | 33 minutes ago
San Diego-inspired legislation that would require superstores to prepare detailed economic analyses before projects can be approved barely escaped the Assembly Local Government Committee Wednesday. It was rescued from the shelf by a last-second...

Pulitzer Prize-winning Jose Antonio Vargas - I'm an illegal immigrant to "...
International Business Times | about 2 hours ago
One August morning nearly two decades ago, my mother woke me and put me in a cab. She handed me a jacket. "Baka malamig doon" were among the few words she said. ("It might be cold there.") When I arrived at the Philippines' Ninoy Aquino...

Pulitzer Winner Says He's an Illegal Immigrant
Fox | about 4 hours ago
This undated handout photo provided by Define American shows Jose Antonio Vargas, a Pulitzer Prize-winning journalist. A Pulitzer Prize-winning journalist who covered presidential politics and the 2007 Virginia Tech shootings for The Washington Post...

Ex-DC journalist says he's an illegal immigrant
AP | Politics | Updated 6 hours ago

NEWS STORIES RELATED TO:

Pulitzer Prize-Winning Reporter is an Illegal Immigrant

by [catspirit](#)

Washington : DC : USA | about 2 hours ago

Journalist Jose Antonio Vargas, winner of the Pulitzer Prize for his part in reporting about the Virginia Tech shootings, has announced that he is an illegal immigrant from the Philippines....

SHARE: [Facebook](#) [Twitter](#) [LinkedIn](#) [StumbleUpon](#) [Reddit](#) [Dribbble](#) [Delicious](#) [Diigo](#) [VK](#) [Odnoklassniki](#) [LiveJournal](#) [MySpace](#) [Flickr](#) [YouTube](#) [Vimeo](#) [SoundCloud](#) [Podcast](#) [RSS](#) [Email](#) [Print](#) [10 retweet](#)

Like [Send](#) [Be the first of your friends to like this.](#)

[Read full report](#)

Reach [Credibility](#) [\[Progress Bar\]](#)

READ MORE: journalist, illegal alien, pulitzer prize winner, José Antonio Vargas, Immigrant, The DREAM Act, Post, dream act, illegal immigration, Pulitzer prize, immigration

Gay Pulitzer Prize-Winning Reporter Jose Antonio Vargas Comes Out as ...
Towleroad | about 10 hours ago

Gay Pulitzer Prize-Winning Reporter Jose Antonio Vargas Comes Out as Undocumented Immigrant Jose Antonio Vargas, a gay journalist who won a Pulitzer Prize for his coverage of the Virginia Tech shootings in the Washington Post

np [[NNP-pulitzer JJ-prize-winning NN-reporter], [JJ-* NN-immigrant]]

Fig. 7. a. News articles and aggregated images found on the web and determined to be relevant to this article. b. Syntactic generalization result for the seed articles and the other article mined for on the web.

above tasks of the sensible/meaningless classes. At the same time, the recall of the former is lower than that of the latter, and the resultant F-measure is slightly higher for the ad lines information extraction, even though the complexity of problem is significantly lower. This can be explained by the relatively high variability of the acceptable ad lines ('sales pitches') that have not been captured by the training set.

The overall recognition accuracy of the epistemic state classification is higher than that of the other two domains because manually built templates for particular states cover a significant portion of the cases. At the same time, the recognition accuracy for particular epistemic states varies significantly from state to state and was mostly determined by how well the various phrasings were covered in the training dataset. For the evaluation dataset, we recognized which epistemic state existed in each of 200 sentences. Frequently, there were two or more such states (without contradictions) per sentence. Note also that epistemic states can overlap. A low classification accuracy occurs when the classes are defined approximately and the boundaries between them are fuzzy and beyond expression in natural language. Therefore, we observe that syntactic generalization provides some semantic cues that would be hard to obtain at the level of keywords or superficial parsing.

6.2. Commercial evaluation of text similarity improvement

We subject the proposed technique of taxonomy-based and syntactic generalization-based techniques to commercial mainstream news analysis at AllVoices.com. The task is to cluster relevant news items together by means of text relevance analysis. By definition, multiple news articles belong to the same cluster if there is a substantial overlap between the involved entities, such as geographical locations, the names of individuals, organizations and other agents, and the relationships between them. Some of these can be extracted using entity taggers and/or taxonomies built offline, and some are handled in real time using syntactic generalization (the bottom of Fig. 7b). The latter is applicable if there is a lack of prior entity information.

In addition to forming a cluster of relevant documents, it is necessary to aggregate relevant images and videos from different sources, such as Google Image, YouTube and Flickr, and access their relevance given their textual descriptions and tags. A similar taxonomy and syntactic generalization-based technique is applied in this process (Fig. 7a).

The precision of the text analysis is achieved by the site's usability (click rate): more than nine million unique visitors per month. Recall is accessed manually; however, the system needs to find at least a few articles, images and videos for each incoming article. Recall is generally not an issue for web mining and web document analysis. Therefore, it is assumed that there is a high number of articles, images and videos on the web for mining.

Relevance is ensured by two steps. First, we form a query to the image/video/blog search engine API, given an event title and first paragraph and extracting and filtering noun phrases by certain significance criteria. Second, we apply a similarity assessment to the texts returned from images/videos/blogs and ensure that substantial common noun, verb or prepositional sub-phrases can be identified between the seed events and these media.

The precision data for the relevance relationships between an article and other articles, blog postings, images and videos are presented in Table 4. Note that by itself, the taxonomy-based method has a very low precision and does not outperform the baseline of the statistical assessment. However, there is a noticeable improvement in the precision of the hybrid system, where the major contribution of syntactic generalization is improved by a few percentage points by the taxonomy-based method [15,57]. We can conclude that syntactic generalization and the taxonomy-based methods (which also rely on syntactic generalization) use different sources of relevance information. Therefore, they are complementary to each other.

The objective of syntactic generalization is to filter out false-positive relevance decisions made by a statistical relevance engines. This statistical engine has been designed following (Liu & Birnbaum 2007; Liu & Birnbaum 2008). The percentage of false-positive news stories was reduced from 29% to 17% (approximately 30,000 stories/month, viewed by 9 million unique users), and the percentage of false positive image attachment was reduced from 24% to 20% (approximately 3000 images and 500 videos attached to stories monthly). The percentages shown are (100% – precision values); recall values are not as important for web mining, assuming there is an unlimited number of resources on the web and that we must identify the relevant ones.

The accuracy of our structural machine learning approach is worth comparing with the other parse tree learning approach based on the statistical learning of SVM. Moschitti [39,42] compares the performances of the bag-of-words kernel, syntactic parse trees and predicate argument structures kernel, and the semantic role kernel, confirming that the accuracy improves in this order and reaches an F-measure of 68% on the TREC dataset. Also, according to Qian et al. [49], unified parse and semantic tree significantly outperforms the single syntactic parse tree, mostly due to a contributions from entity-related semantic features such

Table 4
Improvement in the precision of text similarity.

Media/ method of text similarity assessment	Full size news articles	Abstracts of articles	Blog posting	Comments	Images	Videos
Frequencies of terms in documents (baseline)	29.3%	26.1%	31.4%	32.0%	24.1%	25.2%
Syntactic generalization	19.7%	18.4%	20.8%	27.1%	20.1%	19.0%
Taxonomy-based	45.0%	41.7%	44.9%	52.3%	44.8%	43.1%
Hybrid Syntactic generalization and Taxonomy-based	17.2%	16.6%	17.5%	24.1%	20.2%	18.0%

as its type, subtype, and their bi-gram combinations. A good performance in semantic relation extraction is achieved via a composite kernel, which combines this tree kernel with a linear feature-based kernel.

Achieving comparable accuracies, the kernel-based approach requires manual adjustment; however, it does not provide similarity data in the explicit form of common sub-phrases. Structural machine learning methods are better suited for performance-critical production environments serving hundreds millions of users because they better fit modern software quality assurance methodologies. Logs of the discovered commonality expressions are maintained and tracked, which ensures the required performance as the system evolves over time and the text classification domains change.

7. Related work

Most of the work on automated semantic inference from syntax deals with much lower semantic levels than the semantic classes we manage in this study. de Salvo Braz et al. [21] present a principled, integrated approach to *semantic entailment*. These authors developed an expressive knowledge representation that provides a hierarchical encoding of the structural, relational and semantic properties of the text and populated it using a variety of machine learning-based tools. An inferential mechanism over a knowledge representation that supports both abstractions and several levels of representations allowed them to begin to address important issues in abstracting over the variability in natural language. Certain reasoning patterns from this work are implicitly implemented in the parsing tree matching approach proposed in the current study.

Note that the set of semantic problems addressed in this paper is of a much higher semantic level than the SRL; a more sensitive tree-matching algorithm is required for this semantic level. In terms of this study, the semantic level of classification classes is much higher than the level of semantic role labeling or semantic entailment. SLR does not aim to produce complete formal meanings, in contrast to our approach. Our classification classes, such as proper phrase extraction and relevant/irrelevant search results, are of a high semantic level. However, they cannot be fully formalized; it is difficult to verbalize their criteria, even for human experts.

Usually, classical approaches to semantic inference rely on complex logical representations. However, practical applications usually adopt shallower lexical or lexical-syntactic representations and lack a principled inference framework. Bar-Haim et al. [2] proposed a generic semantic inference framework that operates directly on syntactic trees. New trees are inferred by applying entailment rules, which provide a unified representation for varying types of inferences. Rules are generated by manual and automatic methods, covering generic linguistic structures with specific lexical-based inferences. The current work deals with syntactic tree transformation in the graph-learning framework (compare with refs. [48,55]), treating various phrasings of the same meaning in a more unified and automated manner.

Traditionally, semantic parsers are constructed manually or are based on manually constructed semantic ontologies, but these approaches are delicate and costly. A number of supervised learning approaches to building formal semantic representations have been proposed [65,66]. Unsupervised approaches have been proposed as well; however, they applied only to shallow semantic tasks (e.g., paraphrasing [22], information extraction [6], and semantic parsing [67]). The problem domain in the current study required a much more complex treatment of syntactic peculiarities to perform their classification into semantic classes. In terms of learning, our approach is closer in merit to the unsupervised learning of complete formal semantic representations. Compared to semantic role labeling [11] and other forms of shallow semantic processing, our approach maps text to formal meaning representations obtained via generalization.

Supervised semantic parsing addresses the issue of syntactic variation by learning to construct the grammar automatically from sample meaning annotations [66]. The existing approaches differ in the meaning representation languages they use and the amount of annotation required. In the approach proposed by (Zettlemoyer and Collins, 2005), the training data consists of sentences paired with their meanings in lambda form. A probabilistic combinatory categorical grammar (PCCG) is learned using a loglinear model, where the probability of the final logical form L and the meaning-derivation tree T conditioned on the sentence S is

$$P(L, T|S) = 1/z^* \exp(\sum_i w_i f_i(L, T, S)).$$

Here, Z is the normalization constant, and f_i are the feature functions with weights w_i . Candidate lexical entries are generated by a domain-specific procedure based on the target logical forms. The major limitation of supervised approaches is that they require meaning annotations for sentences. Even in a restricted domain, doing this consistently and with high quality requires a nontrivial effort. For unrestricted text, the complexity and subjectivity of annotation makes this approach essentially infeasible; even prespecifying the target predicates and objects is very difficult. Therefore, to learn semantic data beyond limited domains, it is crucial to develop unsupervised methods that do not rely on labeled meanings.

In the past, unsupervised approaches have been applied to some semantic tasks. For example, DIRT [22] learns paraphrases of binary relations based on the distributional similarity of their arguments; TextRunner [6] automatically extracts relational triples in open domains using a self-trained extractor; and SNE applies relational clustering to generate a semantic network from TextRunner triples (Kok and Domingos, 2008). While these systems illustrate the promise of unsupervised methods, the semantic content they extract is nonetheless shallow, and we believe it is insufficient for the benchmarking problems presented in this work.

A number of semantic-based approaches have been suggested for problems similar to the three problems used for evaluation in this work. (Lamberti et al. [54] proposed a relation-based page rank algorithm to augment Semantic Web search engines employing data extracted from user queries and annotated resources. In this approach, relevance is measured as the probability that the retrieved resource actually contains the relationships whose existence was assumed by the user at the time of the query

definition. In this study, we demonstrated how problems such as search result ranking can be solved based on semantic generalizations based on *local* data and limited to queries and hit snapshots.

Statistical learning has been applied to syntactic parse trees as well. Statistical approaches are generally based on stochastic models [50]. Given a model and an observed word sequence, semantic parsing can be viewed as a pattern recognition problem, and statistical decoding can be used to find the most likely semantic representation.

Convolution kernels are an alternative to the explicit feature design that we perform in this paper. They measure the similarity between two syntactic trees in terms of their sub-structures (e.g., [38]). These approaches use embedded combinations of trees and vectors (e.g., all vs. all summation: each tree and vector of the first object are evaluated against each tree and vector of the second object) and have presented optimal results [44,68] in handling the semantic rolling tasks. For example, given the question “What does S.O.S. stand for?”, the following representations are used, in which the different trees are the question parse tree, the bag-of-words tree, the bag-of-POS-tags tree and the predicate argument tree.

1. (SBARQ (WHNP (WP What))(SQ (AUX does)(NP (NNP S.O.S.))(VP (VB stand)(PP (IN for))));
2. (What*)(does*)(S.O.S.*)(stand*)(for*)(?)*;
3. (WP*)(AUX*)(NNP*)(VB*)(IN*)(.)*;
4. (ARG0 (R-A1 (What*))(ARG1 (A1 (S.O.S. NNP)))(ARG2 (rel stand)).

Although statistical approaches will most likely find a practical application, we believe that currently structural machine learning approaches will provide more explicit insight into the important features of syntactic parse trees.

Web-based metrics that compute the semantic similarity between words or terms [36] are complementary to our measure of similarity. A fundamental assumption is made that similarity of context implies similarity of meaning. Relevant web documents are downloaded via a web search engine, and the contextual information of the words of interest is compared (context-based similarity metrics). This shows that context-based similarity metrics significantly outperform co-occurrence-based metrics in terms of their correlation with human judgment.

8. Conclusions

In this study, we demonstrated that high-level sentences semantic features such as *being informative* can be learned from the low-level linguistic data of a complete parse tree. Unlike the traditional approaches to the *multilevel* derivation of semantics from syntax, we explored the possibility of linking low-level but detailed syntactic levels with high-level pragmatic and semantic levels *directly*.

In recent decades, most approaches to NL semantics relied on mapping to First Order Logic representations with a general prover and without the use of acquired rich knowledge sources. Significant developments in NLP, particularly the ability to acquire knowledge and induce some level of abstract representation, are expected to support more sophisticated and robust approaches. A number of recent approaches are based on shallow representations of text that capture lexico-syntactic correlations based on dependency structures and are mostly built from grammatical functions extending keyword matching [32]. Semantic information, such as WordNet's lexical chains [24], can slightly enrich this representation. Learning various logic representations [33] is reported to improve accuracy as well. de Salvo Braz et al. [21] make global use of a large number of resources and attempts to develop a flexible, hierarchical representation and an inference algorithm. However, we believe that neither of these approaches reaches the high semantic level required for practical application.

Zanzotto et al. [41], Moschitti et al. [40] proposed several kernel functions to model parse tree properties in kernel-based machines, such as perceptrons or support vector machines (see also [35]). In this study, instead of tackling a high-dimensional space of features formed from syntactic parse trees, we apply a more structural machine learning approach to learn syntactic parse trees themselves, measuring similarities via sub-parse trees and not distances in this space. Moschitti et al. define different types of tree kernels as general approaches to feature engineering for semantic role labeling (SLR) and experiment with these kernels to investigate their contributions to the individual stages of an SRL architecture in both isolation and combination with other traditional manually coded features. The results for the boundary recognition, classification, and re-ranking stages provide systematic evidence of the significant impact of tree kernels on the overall accuracy, especially when the amount of training data is small. The structure-based methods used in this study can leverage a limited amount of training cases as well.

Parsing and chunking (conducted by OpenNLP) and syntactic generalization are significantly slower than other operations in a content management system and are comparable with operations like duplicate search. To verify relevance, the application of syntactic generalization should be preceded by statistical keyword-based methods. In real-time application components such as searches, we use a conventional TF*IDF-based approach (such as SOLR/lucene) to find a set of up to 100 candidate answers from millions of documents and then apply syntactic generalization to each candidate. For off-line components, we use parallelized map/reduce jobs (Hadoop) to apply parsing and syntactic generalization to large volumes of data. This approach allowed a successful combination of efficiency and relevance when serving more than 10 million unique site users monthly at Datran.com, allvoices.com, Zvents.com and eBay.com.

In this study, we manually encoded paraphrases for more accurate sentence generalizations. The automated, unsupervised acquisition of paraphrases has been an active research field in recent years, but its effective coverage and performance have rarely been evaluated. Romano et al. [28] proposed a generic paraphrase-based approach for a specific case, such as relation extraction, to obtain a generic configuration for correlations between objects in texts. The need exists for novel robust models that address syntactic complexity and variability when matching paraphrases in texts. We believe the current study is a step in that direction.

Similar to the studies described above, we address semantic inference in a domain-independent manner. At the same time, in contrast to most semantic inference projects, we narrow our focus to a very specific semantic domain (a limited set of classes), solving

a number of practical problems of text classification. The learned structures would vary significantly from one semantic domain to another, in contrast to the general linguistic resources designed for horizontal domains.

Using semantic information for query ranking has been proposed in Aleman-Meza et al. and Ding et al. [51,52], improving search relevance. However, we believe that the current study is a pioneering approach to deriving the semantic information required for ranking from syntactic parse trees *directly*. In our further studies, we plan to proceed from syntactic parse trees to higher semantic levels and to explore applications that would benefit from this work (see also [18]).

Proposed approach is tolerant to errors in parsing. For more complex sentences where parsing errors are likely, using OpenNLP, we select multiple versions of parsings and their estimated confidence levels (probabilities). Then we cross-match these versions and if parsings with lower confidence levels provide a higher match score, we select them.

As to the complexity of syntactic generalization, it can be reduced to constant with respect to the size of parse tree. Computing relation $\Gamma_2 \leq \Gamma_1$ for arbitrary graphs Γ_2 and Γ_1 is an NP-complete problem (since it is a generalization of the subgraph isomorphism problem. Finding $X * Y = Z$ for arbitrary X, Y, and Z is generally an NP-hard problem. A method based on so-called projections was proposed, which allows one to establish a trade-off between accuracy of representation by labeled graphs and complexity of computations with them. In particular, for a fixed size of projections, the worst-case time complexity of computing operation $*$ and testing relation \leq becomes constant. Application of projections was tested in various experiments with chemical (molecular) graphs and conflict graphs [14,16]. As to the complexity of tree kernel algorithms, they can be run in linear average time [40] $O(m + n)$, where m and n are number of nodes in a first and second trees.

As to the future work, we will make a high performance syntactic generalization available for industrial applications as a part of OpenNLP, extend the number of application areas, and extend the model towards multi-sentence discourse.

Acknowledgments

We are grateful to our colleagues SO Kuznetsov, B Kovalerchuk and others for valuable discussions and to our anonymous reviewers for their suggestions. This research is partially funded by the EU Project No. 238887, a unique European Citizens' attention service (ISAC6+) IST-PSP. This research is also funded by the Spanish MICINN (Ministerio de Ciencia e Innovación) IPT-430000-2010-13 project Social powered Agents for Knowledge search Engine (SAKE), TIN2010-17903 Comparative approaches to the implementation of intelligent agents in digital preservation from a perspective of the automation of social networks, and the AGAUR 2011 FLB00927 research grant awarded to Gabor Dóbrocsi and the grup de recerca consolidat CSI-ref.2009SGR-1202.

References

- [1] J.F. Allen, Natural Language Understanding, Benjamin Cummings, 1987.
- [2] R. Bar-Haim, I. Dagan, I. Greental, E. Shnarch, Semantic Inference at the Lexical-Syntactic Level AAAI-05, 2005.
- [3] W. Buntine, Generalized subsumption and its applications on induction and redundancy, Artificial Intelligence 36 (1988) 149–176.
- [4] P. Bille, A Survey on Tree Edit Distance and Related Problems, Journal Theoretical Computer Science Archive 337 (1–3) (2005) 217–239.
- [5] P. Bulychyev, M. Minea, Duplicate code detection using anti-unification, in: IWSC, 2009.
- [6] Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, Oren Etzioni, Open information extraction from the web, in: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence AAAI Press, Hyderabad, India, 2007, pp. 2670–2676.
- [7] U. Baldewein, Katrin Erk, Sebastian Padó, Detlef Prescher, Semantic role labeling with chunk sequences, 2004. CoNLL-2004 Boston, MA.
- [8] M. Dzikovska, M. Swift, J. Allen, William de Beaumont, Generic parsing for multi-domain semantic interpretation, in: International Workshop on Parsing Technologies (Iwpt05), Vancouver BC, 2005.
- [9] K. Hacioglu, S. Pradhan, W. Ward, J.H. Martin, D. Jurafsky, Semantic role labeling by tagging syntactic chunks, in: Proceedings of the Eighth Conference on Computational Natural Language Learning, Proc. of CoNLL-04ACL, Boston, MA, 2004.
- [10] C. Cardie, R.J. Mooney, Machine learning and natural language, Machine Learning 1 (5) (1999).
- [11] X. Carreras, Luis Marquez, Introduction to the CoNLL-2004 shared task: Semantic role labeling, in: Proceedings of the Eighth Conference on Computational Natural Language Learning ACL, Boston, MA, 2004, pp. 89–97.
- [12] B. Galitsky, Natural language question answering system: technique of semantic headers, Advanced Knowledge International, Australia, 2003.
- [13] B. Galitsky, S.O. Kuznetsov, Learning communicative actions of conflicting human agents, Journal of Experimental & Theoretical Artificial Intelligence 20 (4) (2008) 277–317.
- [14] B. Galitsky, M.P. González, C.I. Chesñevir, A novel approach for classifying customer complaints through graphs similarities in argumentative dialogue, Decision Support Systems 46–3 (2009) 717–729.
- [15] B. Galitsky, G. Dobrocsi, J.L. de la Rosa, S.O. Kuznetsov, Using generalization of syntactic parse trees for taxonomy capture on the web, ICCS 2011 (2011) 104–117.
- [16] B. Galitsky, Josep Lluís de la Rosa, Concept-based learning of human behavior for customer relationship management. Special Issue on Information Engineering Applications Based on Lattices, Information Sciences 181 (10) (2011) 2016–2035.
- [17] B. Galitsky, D.A. Dremov, S.O. Kuznetsov, Increasing the relevance of meta-search using parse trees, in: 12th Russian National AI Conference, M., PhysMatLit, vol. 1, 2010, pp. 261–266, (In Russian).
- [18] B. Galitsky, S.O. Kuznetsov, Semantic classification based on machine learning of parse trees, in: 12th Russian National AI Conference, M., PhysMatLit, vol. 1, 2010, pp. 261–266, (In Russian).
- [19] M. Tatu, D. Moldovan, A logic-based semantic approach to recognizing textual entailment, in: Proceedings of the COLING/ACL, 2006.
- [20] V. Punyakanok, D. Roth, W. Yih, The Necessity of Syntactic Parsing for Semantic Role Labeling IJCAI-05, 2005.
- [21] R. de Salvo Braz, R. Girju, V. Punyakanok, D. Roth, M. Sammons, An inference model for semantic entailment in natural language, in: Proc AAAI-05, 2005.
- [22] D. Lin, P. Pantel, DIRT: discovery of inference rules from text, in: Proc. of ACM SIGKDD Conference on Knowledge Discovery and Data Mining 2001, 2001, pp. 323–328.
- [23] J.S. Mill, A system of logic, ratiocative and inductive, Longmans Green, London, 1843.
- [24] D. Moldovan, C. Clark, S. Harabagiu, S. Maiorano, Cogex: a logic prover for question answering, in: Proc. of HLTNAACL 2003, 2003.
- [25] P. Moreda, B. Navarro, M. Palomar, Corpus-based semantic role approach in information retrieval, Data & Knowledge Engineering 61 (2007) 467–483.
- [26] G.D. Plotkin, A note on inductive generalization, in: B. Meltzer, D. Michie (Eds.), Machine Intelligence, vol. 5, Elsevier North-Holland, New York, 1970, pp. 153–163.
- [27] J.A. Robinson, A machine-oriented logic based on the resolution principle, Journal of the Association for Computing Machinery 12 (1965) 23–41.
- [28] L. Romano, M. Kouylekov, I. Szpektor, I. Dagan, A. Lavelli, Investigating a generic paraphrase-based approach for relation extraction, in: Proceedings of EACL, 2006, pp. 409–416.
- [29] J.C. Reynolds, Transformational systems and the algebraic structure of atomic formulas, Machine Intelligence 5 (1) (1970) 135151.

- [30] D. Ravichandran, E. Hovy, Learning surface text patterns for a Question Answering system, in: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002), Philadelphia, PA, 2002.
- [31] M. Stevenson, M.A. Greenwood, A semantic approach to IE pattern induction, in: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005), Ann Arbor, Michigan, 2005.
- [32] B.V. Durme, Y. Huang, A. Kupsc, E. Nyberg, Towards light semantic processing for question answering, HLT Workshop on Text Meaning, 2003.
- [33] C. Thompson, R. Mooney, L. Tang, Learning to parse NL database queries into logical form, in: Workshop on Automata Induction, Grammatical Inference and Language Acquisition, 1997.
- [34] M.H. Sorensen, R. Gluck, An algorithm of generalization in positive supercompilation, in: Logic Programming: Proceedings of the International Symposium, MIT Press, 1995.
- [35] D. Zhou, Y. He, Discriminative training of the hidden vector state model for semantic parsing, IEEE Transactions on Knowledge and Data Engineering (Jan 2009) 21–1.
- [36] E. Iosif, A. Potamianos, Unsupervised semantic similarity computation between terms using web documents, IEEE Transactions on Knowledge and Data Engineering 13 (Oct. 2009).
- [37] K. Williams, Christopher Dozier, Andrew McCulloh, Learning transformation rules for semantic role labeling, in: Proceedings of the Eighth Conference on Computational Natural Language Learning ACL, Boston, MA, 2004, pp. 89–97, CoNLL-2004 Boston MA.
- [38] M. Collins, Nigel Duffy, New ranking algorithms for parsing and tagging: kernels over discrete structures, and the voted perceptron, in: ACL02, 2002.
- [39] A. Moschitti, Efficient convolution kernels for dependency and constituent syntactic trees, in: Proceedings of the 17th European Conference on Machine Learning, Berlin, Germany, 2006.
- [40] Alessandro Moschitti, Kernel Methods, Syntax and Semantics for Relational Text Categorization, in: proceeding of ACM 17th Conference on Information and Knowledge Management (CIKM), Napa Valley, California, 2008.
- [41] F.M. Zanzotto, Alessandro Moschitti, Automatic learning of textual entailments with cross-pair similarities, in: Proceedings of the Joint 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING–ACL), Sydney, Australia, 2006.
- [42] A. Moschitti, Syntactic and semantic kernels for short text pair categorization, in: Proceedings of the 12th Conference of the European Chapter of the ACL, 2009.
- [43] P. Kingsbury, M. Palmer, From Treebank to PropBank, in: Proc. of the 3rd LREC, Las Palmas, 2002.
- [44] A. Moschitti, Daniele Pighin, Roberto Basili, Semantic role labeling via tree kernel joint inference, in: Proceedings of the 10th Conference on Computational Natural Language Learning, New York, USA, 2006.
- [45] H. Mangassarian, Hassan Artail, A general framework for subjective information extraction from unstructured English text, Data & Knowledge Engineering 62 (2) (August 2007) 352–367.
- [46] P. Moreda, M. Palomar, The Role of Verb Sense Disambiguation in Semantic Role Labeling, in: T. Salakoski, F. Ginter, S. Pyysalo, T. Pahakkala (Eds.), FinTAL, LNAL, vol. 4139, Springer Heidelberg, 2006, pp. 684–695.
- [47] P. Domingos, H. Poon, Unsupervised semantic parsing, with, in: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing ACL, Singapore, 2009.
- [48] S. Kapoor, H. Ramesh, Algorithms for enumerating all spanning trees of undirected and weighted graphs, SIAM Journal on Computing 24 (1995) 247–265.
- [49] L. Qian, Guodong Zhou, Qiaoming Zhu, Employing constituent dependency information for tree kernel-based semantic relation extraction between named entities, Journal of ACM Transactions on Asian Language Information Processing 10 (3) (2011) (Article 15 (September 2011), 24 pages).
- [50] M. Zhang, G.D. Zhou, A. Aw, Exploring syntactic structured features over parse trees for relation extraction using kernel methods, Information Processing and Management: An International Journal 44 (2) (March 2008) 687–701.
- [51] B. Aleman-Meza, C. Halaschek, I. Arpinar, A. Sheth, A context-aware semantic association ranking, in: Proc. First Int'l Workshop Semantic Web and Databases (SWDB '03), 2003, pp. 33–50.
- [52] L. Ding, T. Finin, A. Joshi, R. Pan, R.S. Cost, Y. Peng, P. Reddivari, V. Doshi, J. Sachs, Swoogle: a search and metadata engine for the semantic web, in: Proc. 13th ACM Int'l Conf. Information and Knowledge Management (CIKM '04), 2004, pp. 652–659.
- [53] OpenNLP, <http://incubator.apache.org/opennlp/documentation/manual/opennlp.htm> 2012.
- [54] F. Lamberti, Andrea Sanna, Claudio Demartini, A relation-based page rank algorithm for semantic web search engines, IEEE Transactions on Knowledge and Data Engineering 21 (1) (Jan. 2009) 123–136.
- [55] D. Chakrabarti, C. Faloutsos, Graph mining: laws, generators, and algorithms, ACM Computing Surveys, vol. 38, no. 1, 2006.
- [56] S. Abney, Parsing by chunks, in: Principle-Based Parsing, Kluwer Academic Publishers, 1991, pp. 257–278.
- [57] B. Galitsky, Josep Lluís de la Rosa, Gabor Dobrocsi, Building integrated opinion delivery environment, FLAIRS-24, West Palm Beach FL May 2011, 2011.
- [58] D. Gildea, Loosely tree-based alignment for machine translation, in: Proceedings of the 41th Annual Conference of the Association for Computational Linguistics (ACL-03), 2003, pp. 80–87, Sapporo, Japan.
- [59] H. Bunke, Graph-based tools for data mining and machine learning, Machine Learning and Data Mining in Pattern Recognition, Lecture Notes in Computer Science 2734/2003 (2003) 7–19.
- [60] J. Vanhatalo, Hagen Völzer, Jana Koehler, The refined process structure tree, Data & Knowledge Engineering 68 (9) (September 2009) 793–818.
- [61] A. Weichselbraun, Gerhard Wohlgenannt, Arno Scharl, Refining non-taxonomic relation labels with external structured data to support ontology learning, Data & Knowledge Engineering 69 (8) (August 2010) 763–778.
- [62] K. Fukunaga, Introduction to statistical pattern recognition, Academic Press, San Diego, 1990.
- [63] C. Cortes, V. Vapnik, Support-Vector Networks. Machine Learning 20 (1995) 273–297.
- [64] S. Ducheyne, J.S. Mill's Canons of Induction: From true causes to provisional ones, History and Philosophy of Logic 29 (4) (2008) 361–376.
- [65] L.S. Zettlemoyer, M. Collins, Learning to Map Sentences to Logical Form: Structured Classification with Probabilistic Categorical Grammars, In Proceedings of the Twenty First Conference on Uncertainty in Artificial Intelligence (UAI) 2005, 2005.
- [66] Y.W. Wong, R. Mooney, Learning synchronous grammars for semantic parsing with lambda calculus, Annual Meeting-Association for computational Linguistics 45 (1) (2007) 960.
- [67] H. Poon, P. Domingos, Joint unsupervised coreference resolution with Markov logic, In Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing 2008, Honolulu, HI. ACL, 2008649658[68] A. Moschitti A study on convolution kernels for shallow semantic parsing, Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics 2004335



Boris Galitsky has been contributing natural language-related technologies to Silicon Valley, USA start-ups over last two decades. In 1999 he co-founded iAskWeb which was providing tax and investment recommendations to customers of a few Fortune 500 companies. He contributed his linguistic technology to Xoopit, acquired by Yahoo, Uptake, acquired by Groupon, and LogLogic, acquired by Tibco. He received his PhD in natural language understanding in 1994 and ANECA/EU Associate Professorship degree in 2011. Boris authored more than 70 publications, a book and multiple patents in the field of natural language understanding. Boris is currently a lead scientist at eBay.



Prof. Josep Lluís de la Rosa, peplluis@eia.udg.edu, h-index = 15, MSc and Ph.D. in Computer Engineering from the Autonomous University of Barcelona (UAB), Barcelona, in 1989 and 1993, MBA in 2002. He is professor of the Universitat de Girona (UdG) and director of the ARLab (Agents Research Laboratory – GRCT69). He has published more than 100+ papers in international journals and 300+ papers in international conferences, 4 patents and 3 spin-off companies. He was visiting professor at Rensselaer Polytechnic Institute (RPI) in 2008–2010. His research interests focus on intelligent agents, understanding the agency property of introspection or self-awareness, as well as understanding its impact in the emergent behaviour of billions of agents by means of the computational ecologies models. Digital preservation, social networks and complementary currencies are the areas of application. He has participated in several successful EU projects like ONE, Open Negotiation Environments FP6-2005-IST-5, grant agreement num. 34744 (2006–2009) and PROTAGE Preservation of Digital Information with Intelligent Agents, (2007–2011).



Gábor Dobrocsi is an Informatics Engineer who earned his M.Sc. at the University of Miskolc (Hungary). After he received his degree he became a visitor scientist at the Rensselaer Polytechnic Institute (USA) where he participated in an academic research to develop an alternative review system for scientific publications. Then he joined to the development team of a high end commercial citizens' information and assessment service system featuring natural language processing techniques at EASY Innova (Spain). Currently he is a PhD student at the University of Girona (Spain) and researching on the fields of Agent technologies, Social search and recommendation systems and natural language processing.