UTD: Ensemble-Based Spatial Relation Extraction

Jennifer D'Souza and **Vincent Ng**
Human Language Technology Research Institute
University of Texas at Dallas
Richardson, TX 75083-0688
`{jld082000,vince}@hlt.utdallas.edu`

Abstract

SpaceEval (SemEval 2015 Task 8), which concerns spatial information extraction, builds on the spatial role identification tasks introduced in SemEval 2012 and used in SemEval 2013. Among the host of subtasks presented in SpaceEval, we participated in subtask 3a, which focuses solely on spatial relation extraction. To address the complexity of a MOVELINK, we decompose it into smaller relations so that the roles involved in each relation can be extracted in a joint fashion without losing computational tractability. Our system was ranked first in the official evaluation, achieving an overall spatial relation extraction F-score of 84.5%.

1 Introduction

SpaceEval[1] was organized as a shared task for the semantic evaluation of spatial information extraction (IE) systems. The goals of the shared task include identifying and classifying particular constructions in natural language for expressing spatial information that are conveyed through the spatial concepts of locations, entities participating in spatial relations, paths, topological relations, direction and orientation, motion, etc. It presents a wide spectrum of spatial IE related subtasks for interested participants to choose from, building on the two previous years shared tasks on the same topic (Kordjamshidi et al., 2012; Kolomiyets et al., 2013).

Our goal in this paper is to describe the version of our spatial relation extraction system that partic-

ipated in subtask 3a of SpaceEval. Systems participating in this subtask assume as input the *spatial elements* in a text document. For example, in the sentence *The flower is in the vase$_1$ and the vase$_2$ is on the table*, the set of spatial elements {*flower, in, vase$_1$, vase$_2$, on, table*} are given and subsequently used as candidates for predicting spatial relations. Leveraging the successes of a joint role-labeling approach to spatial relation extraction involving stationary objects, we employ it to extract so-called MOVELINKs, which are spatial relations defined over objects in motion. In particular, we discuss the adaptations needed to handle the complexity of MOVELINKs. Experiments on the SpaceEval corpus demonstrate the effectiveness of our ensemble-based approach to spatial relation extraction. Among the three teams participating in subtask 3a, our team was ranked first in the official evaluation, achieving an overall F-score of 84.5%.

The rest of the paper is organized as follows. We first give a brief overview of the subtask 3a of SpaceEval and the corpus (Section 2). After that, we describes related work (Section 3). Finally, we present our approach (Section 4), evaluation results (Section 5), and conclusions (Section 6).

2 The SpaceEval Task

2.1 Subtask 3a: Task Description

Subtask 3a focuses solely on spatial relation extraction using a specified set of spatial elements for a given sentence. Specifically, given an n-tuple of participating entities, the goal is to (1) determine whether the entities in the n-tuple form a spatial re-

---

[1]http://alt.qcri.org/semeval2015/task8/

lation, and if so, (2) classify the roles of each participating entity in the relation.

2.2 Training Corpus

To facilitate system development, 59 travel narratives are marked up with seven types of spatial elements (Table 1) and three types of spatial relations (Table 2), following the ISO-Space (Pustejovsky et al., 2013) annotation specifications, and provided as training data. Note that a *spatial-signal* entity has a semantic-type attribute expressing the type of the relation it triggered. Its semantic-type can be topological, directional, or both.[2]

What is missing in Table 2 about spatial relations is that each entity participating in a relation has a *role*. In QSLINKs and OLINKs, an element can participate as a $trajector$ (i.e., object of interest), $landmark$ (i.e., the grounding location), or $trigger$ (i.e., the relation indicator). Thus the QSLINK and OLINK examples shown in Table 2, are actually represented as the triplet (flower$_{trajector}$, vase$_{landmark}$, in$_{trigger}$). While QSLINK and OLINK relations can have only three fixed participants, a MOVELINK relation has two fixed participants and up to six optional participants to capture more precisely the relational information expressed in the sentence. The two mandatory MOVELINK participants are a $mover$ (i.e., object in motion), and a $trigger$ (i.e., verb denoting motion). The six optional MOVELINK participants are: $source$, $midpoint$, $goal$, $path$, and $landmark$, express different aspects of the $mover$ in space, whereas a $motion\text{-}signal$ connects the spatial aspect to the $mover$.

Note that all spatial relations are *intra-sentential*. In other words, all spatial elements participating in a relation must appear in the same sentence.

3   Related Work

Recall from Section 2 that spatial relation extraction is composed of two subtasks, *role labeling* and *relation classification* of spatial elements. Prior systems have adopted either a *pipeline* approach or a *joint* approach to these subtasks. Given an n-tuple of distinct spatial elements in a sentence, a pipeline spatial

relation extraction system first assigns a role to each spatial element and then uses a binary classifier to determine whether the elements form a spatial relation or not (Kordjamshidi et al., 2011; Bastianelli et al., 2013; Kordjamshidi and Moens, 2014).

One weakness of pipeline approaches is that errors in role labeling can propagate to the relation classification component. To address this problem, *joint* approaches were investigated (Roberts and Harabagiu, 2012; Roberts et al., 2013). Given an n-tuple of distinct spatial elements in a sentence with an assignment of roles to each element, a joint spatial relation extraction system uses a binary classifier to determine whether these elements form a spatial relation with the roles correctly assigned to all participating elements. In other words, the classifier will label the n-tuple as TRUE if and only if (1) the elements in the n-tuple form a relation and (2) their roles in the relation are correct.

We conclude this section by noting that virtually all existing systems were developed on datasets that adopted different or simpler representations of spatial information than SpaceEval's ISO-Space (2013) representation (Mani et al., 2010; Kordjamshidi et al., 2010; Kordjamshidi et al., 2012; Kolomiyets et al., 2013). In other words, none of these systems were designed to identify MOVELINKs.

4   Our Approach

To avoid the error propagation problem, we perform *joint* role labeling and relation extraction. Unlike previous work, where a single classifier was trained, we employ an ensemble of eight classifiers. Creating the eight classifiers permits (1) separating the treatment of MOVELINKs from QSLINKs and OLINKs; and (2) simplifying MOVELINK extraction.

We separate MOVELINKs from QSLINKs and OLINKs for two reasons. First, MOVELINKs involve objects in motion, whereas the other two link types involve stationary objects. Second, MOVELINKs are more complicated than the other two link types: while QSLINKs and OLINKs have three fixed participants, $trajector$, $landmark$ and $trigger$, MOVELINKs can have up to eight participants, including two mandatory participants (i.e., $mover$ and $trigger$) and six optional participants (i.e., $source$, $midpoint$, $goal$, $path$, $landmark$,

---

[2]In the ISO-Space scheme (Pustejovsky et al., 2013), different spatial entities have different attributes. We omit their description here owing to space limitations.

place (e.g., Rome)	*path* (e.g., road)	*spatial-entity* (e.g., car)	*non-motion* event (e.g., is "serving")	*motion* event (e.g., arrived)	*motion-signal* (e.g., by car)	*spatial-signal* (e.g., north of)

Table 1: Seven types of spatial elements in SpaceEval.

Relation	Description	Total
QSLINK	Exists between stationary spatial elements with a regional connection. E.g., in *The flower is in the vase*, the region of the *vase* has an internal connection with the region of the *flower* and hence they are in a QSLINK.	968
OLINK	Exists between stationary spatial elements expressing their relative or absolute orientations. E.g., in *The flower is in the vase*, the *flower* and the *vase* also have an OLINK relation conveying that the *flower* is oriented inside the *vase*.	244
MOVELINK	Exists between spatial elements in motion. E.g., the sentence *He biked from Cambridge to Maine* has a MOVELINK between mover *He*, motion verb *biked*, source of motion *Cambridge*, and goal of motion *Maine*.	803

Table 2: Three spatial relation types in SpaceEval. The "Total" column shows the number of instances annotated with the corresponding relation in the training data.

and *motion-signal*). Given the complexity of a MOVELINK, we decompose a MOVELINK into a set of simpler relations that are to be identified by an ensemble of classifiers.

In the rest of this section, we describe how we train and test our ensemble.

4.1 Training the Ensemble

We employ one classifier for identifying QSLINK and OLINK relations (Section 4.1.1) and seven classifiers for identifying MOVELINK relations (Section 4.1.2).

4.1.1 The LINK Classifier

We collapse QSLINKs and OLINKs to a single relation type, LINK, identifying these two types of links using the LINK classifier. To understand why we can do this, first note that in QSLINKs and OLINKs, the *trigger* has to be a *spatial-signal* element having a semantic-type attribute. If its semantic-type is topological, it triggers a QSLINK; if it is directional, it triggers an OLINK; and if it is both it triggers both relation types. Hence, if a LINK is identified by our classifier, we can simply use the semantic-type value of the relation's *trigger* element to automatically determine whether the relation is a QSLINK an OLINK, or both.

We create training instances for training a LINK classifier as follows. Following the joint approach described above, we create one training instance for each possible role labeling of each triplet of distinct spatial elements in each sentence in a training document. The role labels assigned to the spatial elements in each triplet are subject to the following constraints: (1) each triplet contains a *trajector*, a *landmark*, and a *trigger*; (2) neither the *trajector* nor the *landmark* are of type *spatial-signal* or *motion-signal*; and (3) the *trigger* is a *spatial-signal*.[3] Note that these role constraints are derived from the data annotation scheme. It is worth noting that while we enforce such global role constraints when creating training instances, Kordjamshidi and Moens (2014) enforce them at inference time using Integer Linear Programming.

A training instance is labeled as TRUE if and only if the elements in the triplet form a relation and their roles in the relation are correct. As an example, for the QSLINK and OLINK sentence in Table 2, exactly one positive instance, LINK(flower$_{trajector}$, vase$_{landmark}$, in$_{trigger}$), will be created.

Each instance is represented using the 31 features shown in Table 3. These features are modeled after those employed by state-of-the-art spatial rela-

---

[3] LINKs can have at most one *implicit* spatial element. For example, the sentence *The balloon went up* has LINK(balloon$_{trajector}$, on$_{trigger}$) with an implicit *landmark*. To account for LINKs with implicit *trajector*, *landmark*, or *trigger* participants, we generate three additional triplets from each LINK triplet, one for each participant having the value IMPLICIT.

864

1. **Lexical** (6 features)

    1. concatenated lemma strings of e_1, e_2, and e_3
    2. concatenated word strings of e_1, e_2, and e_3
    3. lexical pattern created from e_1, e_2, and e_3 based on their order in the text (e.g., $Trajector\_is\_Trigger\_Landmark$)
    4. words between the spatial elements
    5. e_3's words
    6. whether e_2's phrase was seen in role r_3 in the training data

2. **Grammatical** (5 features)

    1. dependency paths from e_1 to e_3 to e_2 obtained using the Stanford Dependency Parser (de Marneffe et al., 2006)
    2. dependency paths from e_1 to e_2
    3. dependency paths from e_3 to e_2
    4. paths from e_3 to e_2 concatenated with e_3's string
    5. whether e_1 is a prepositional object of a preposition of an element posited in role r_3 in any other relation

3. **Semantic** (9 features)

    1. WordNet (Miller, 1995) hypernyms and synsets of e_1/e_2
    2. semantic role labels of $e_1/e_2/e_3$ obtained using SENNA (Collobert et al., 2011)
    3. General Inquirer (Stone et al., 1966) categories shared by e_1 and e_2
    4. VerbNet (Kipper et al., 2000) classes shared by e_1 and e_2

4. **Positional** (2 features)

    1. order of participants in text (e.g., r_2-r_1-r_3)
    2. whether the order is r_3-r_2-r_1

5. **Distance** (3 features)

    1. distance in tokens between e_1 and e_3 and that between e_2 and e_3
    2. using a bin of 5 tokens, the concatenated binned distance between (e_1,e_2), (e_1,e_3), and (e_2,e_3)

6. **Entity attributes** (3 features)

    1. spatial entity type of $e_1/e_2/e_3$

7. **Entity roles** (3 features)

    1. predicted spatial roles of $e_1/e_2/e_3$ obtained using our in-house relation role labeler

Table 3: 31 features for spatial relation extraction. Each training instance corresponds to a triplet (e_1,e_2,e_3), where e_1, e_2, and e_3 are spatial elements of types t_1, t_2, and t_3, with participating roles r_1, r_2, and r_3, respectively.

tion extraction systems. Recall that these systems were developed on datasets that adopted different or simpler representations of spatial information than SpaceEval's ISO-Space (2013) representation (Mani et al., 2010; Kordjamshidi et al., 2010; Kordjamshidi et al., 2012; Kolomiyets et al., 2013). Hence, these 31 features have not been used to train classifiers for

extracting MOVELINKs.

We train the LINK classifier using the SVM learning algorithm as implemented in the SVM^{light} software package (Joachims, 1999). To optimize classifier performance, we tune two parameters, the regularization parameter C (which establishes the balance between generalizing and overfitting the classi-

fier model to the training data) and the cost-factor parameter J (which outweighs training errors on positive examples compared to the negative examples), to maximize F-score on development data.

4.1.2 The Seven MOVELINK Classifiers

If we adopted the aforementioned joint method as is for extracting MOVELINKs, each instance would correspond to an octuple of the form: MOVELINK($trigger_i$, $mover_j$, $source_k$, $mid\text{-}point_m$, $goal_n$, $landmark_o$, $path_p$, $motion\text{-}signal_r$), where each participant in the octuple is either a distinct spatial element with a role or the NULL element (if it is not present in the relation). However, generating role permutations for octuples from all spatial elements in a sentence is computationally infeasible. In order to address this tractability problem, we simplify MOVELINK extraction as follows. First, we decompose the MOVELINK octuple into seven smaller tuples including one pair and six triplets. The seven tuples are: (i) ($trigger_i$, $mover_j$); (ii) ($trigger_i$, $mover_j$, $source_k$); (iii) ($trigger_i$, $mover_j$, $mid\text{-}point_m$); (iv) ($trigger_i$, $mover_j$, $goal_n$); (v) ($trigger_i$, $mover_j$, $landmark_o$); (vi) ($trigger_i$, $mover_j$, $path_p$); (vii) ($trigger_i$, $mover_j$, $motion\text{-}signal_r$). Then, we create seven separate classifiers for identifying the seven MOVELINK tuples, respectively.

Using this decomposition for MOVELINK instances, we can generate instances for each classifier using the aforementioned joint approach as is. For instance, to train classifier (i), we generate candidate pairs of the form ($trigger_i$, $mover_j$), where $trigger_i$ and $mover_j$ are spatial elements proposed as a candidate *trigger* and *mover*, respectively. Positive training instances are those ($trigger_i$, $mover_j$) pairs annotated with a relation in the training data, while the rest of the candidate pairs are negative training instances. The instances for training the remaining six classifiers are generated similarly.

As in the LINK classifier, we enforce global role constraints when creating training instances for the MOVELINK classifiers. Specifically, the roles assigned to the spatial elements in each training instance of each of the MOVELINK classifiers are subject to the following constraints: (1) the *trigger* has type $motion$; (2) the *mover* has type $place$, $path$, $spatial\text{-}entity$ or $non\text{-}motion$ event; (3) the *source*, the *goal*, and the *landmark* can be NULL

or has type $place$, $path$, $spatial\text{-}entity$, or $non\text{-}motion$ event; (4) the *mid-point* can be NULL or has type $place$, $path$, or $spatial\text{-}entity$; (5) the *path* can be NULL or has type $path$; and (6) the *motion-signal* can be NULL or has type $motion\text{-}signal$.

Our way of decomposing the octuple along roles can be justified as follows. Since the shared task evaluates MOVELINKs only based on its mandatory *trigger* and *mover* participants, we have a classifier for classifying this core aspect of a motion relation. The next six classifiers, (ii) to (vii), aim to improve the core MOVELINK extraction by exploiting the stronger contextual dependencies with each of its unique spatial aspects namely the *source*, the *mid-point*, the *goal*, the *landmark*, the *path*, and the *motion-signal*.

As an example, for the MOVELINK sentence in Table 2, we will create three positive instances: ($He_{trigger}$, $biked_{mover}$) for classifier (i), ($He_{trigger}$, $biked_{mover}$, $Cambridge_{source}$) for classifier (ii), and ($He_{trigger}$, $biked_{mover}$, $Maine_{goal}$) for classifier (iv).

We represent each training instance using the 31 features shown in Table 3, and train each of the MOVELINK classifiers using SVM^{light}, with the C and J values tuned on development data.

4.2 Testing the Ensemble

After training, we apply the resulting classifiers to classify the test instances, which are created in the same manner as the training instances. As noted before, the LINK spatial relations extracted from a test document by the LINK classifier are further qualified as QSLINK, OLINK, or both based on the semantic-type attribute value of its $trigger$ participant. The MOVELINK relations are extracted from a test document by combining the outputs from the seven MOVELINK classifiers. We explore three different ways of combining the outputs. The first way is simply to combine the outputs from all seven classifiers. However, combining outputs in this way could produce erroneous MOVELINK results, because it could result in a spatial element being classified with more than one role in the same relation since the classifications are made independently. To address this problem, we adopt a second way of combining the seven classifier outputs to generate MOVELINKs. Our second approach resolves multiple role classifications for the same element in a relation by se-

| | QSLINK | | | | | | OLINK | | | | | | MOVELINK | | | | | | OVERALL | | |
| | False | | | True | | | False | | | True | | | False | | | True | | | | | |
	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F
Training Data	99.5	99.4	99.5	46.9	48.9	47.9	100	99.4	99.7	50.3	100	66.9	91.3	99.8	95.3	84.8	61.5	71.3	78.8	84.8	80.1
Test Data	99.6	99.3	99.4	55.3	68	61	99.9	99.9	99.9	67.9	76	71.7	98.3	97.3	97.8	72.7	81.6	76.9	82.3	87	84.5

Table 4: Results for spatial relation extraction using gold spatial elements.

lecting the role that was predicted with highest confidence by the SVM. Our third approach addresses this problem, alternatively, by using a predetermined precedence of roles, decided based on training data statistics of roles' frequency, and selecting the role that appears more frequently in the training data than the other classified roles. Evaluations of the respective outputs produced by adopting each of these three ways showed that they all achieved a very similar level of performance.

5 Evaluation

In this section, we evaluate our ensemble approach to spatial relation extraction.

5.1 Experimental Setup

Dataset. We use the 59 travel narratives released as the SpaceEval challenge training data for system training and development. For testing, we use the 16 travel narratives released as the SpaceEval challenge test data.

Evaluation metrics. Evaluation results are obtained using the official SpaceEval challenge scoring program. Results are expressed in terms of recall (R), precision (P), and F-score (F). When computing recall and precision, true positives for QSLINKs and OLINKs are those extracted ($trajector,landmark,trigger$) triplets that match with those in the gold data. True positives for MOVELINKs are those extracted ($trigger,mover$) pairs found in the gold data.[4]

Parameter tuning. As mentioned in the previous section, we tune the C and J parameters on development data when training each SVM classifier.

---

[4]During system development, we observed that ($trigger,mover$) extraction can be improved by exploiting its stronger dependencies with the optional MOVELINK participants. Therefore, we have classifiers (ii) to (vii) in our ensemble for extracting ($trigger,mover$) pairs missed by classifier (i).

More specifically, during system training and development, we perform five-fold cross validation. In each fold experiment, we use three folds for training, one fold for development, and one fold for testing.

Since joint tuning of these two parameters are computationally expensive, we tune them as follows. We first tune C by setting the J parameter to the default value in SVM^{light}. After finding the C parameter that maximizes F-score on the development set, we fix C and tune J to maximize F-score on the development set.[5]

5.2 Results and Discussion

Table 4 shows the spatial relation extraction results using gold spatial elements of our classifier ensemble from the official SpaceEval scoring program.

The first row shows results from five-fold cross validation on the training data. In each fold experiment, we first tune the learning parameters of each classifier as described in Section 5.1, and then retrain the classifier on all four folds using the learned parameters before applying it to the test fold. The results reported are averaged over the five test folds. The second row results are obtained from evaluation on the official test data. Here, we train each classifier on all of the training data. The learning parameters of each classifier are tuned based on cross validation on the training data. Specifically, we select the parameters that give the best averaged F-score over the five development folds described in Section 5.1.

The column-wise results in the table show performance on extracting the QSLINK, OLINK, and MOVELINK spatial relations types, respectively, and overall. The results under column "False" for each relation type show performance in rejecting the relation candidates that are not actual relations in the gold data. And the results under column "True" for

---

[5]For parameter tuning, C is chosen from the set {0.01,0.05,0.1,0.5,1.0,10.0,50.0,100.0} and J is chosen from the set {0.01,0.05,0.1,0.5,1.0,2.0,4.0,6.0}.

	R	P	F
Training Data	60.7	70.1	62
Test Data	65.3	75.2	69.9

Table 5: Overall results for spatial relation extraction of "True" relations using gold spatial elements.

each relation type show performance in extracting relation candidates that are actual relations in the gold data.

From Table 4, we see that on both the training and test data, performance on rejecting the False relation candidates is close to 100%. However, performance on extracting the True relations is relatively much lower. In decreasing order of performance, our approach is most effective on extracting MOVELINKs, followed by OLINKs, and then QS-LINKs. Thus the relation types on which our approach performs poorly can direct our future efforts in improving performance on this task. We see close to 80% overall relation extraction F-score of our system on both training and test data. This high performance is mainly owing to the high performance of our approach in rejecting the False relation candidates. To better reflect the overall performance of our approach, we show in Table 5 our overall results in extracting True relation types using only the results in "True" columns of Table 4 for the three relation types. From these results, we see that our system performance is in the range of 65-70% F-score on extracting the "True" spatial relations in both datasets. Thus we see that there is still more scope for improvement of our system in order to make it practically usable for spatial relation extraction.

6 Conclusion

We employed an ensemble approach to spatial relation extraction. To address the complexity of a MOVELINK, we decomposed it into smaller relations so that the roles involved in each relation could be extracted in a joint fashion without losing computational tractability. When evaluated on the SpaceEval official test data for subtask 3a, our approach was ranked first, achieving an F-score of 84.5%.

References

Emanuele Bastianelli, Danilo Croce, Daniele Nardi, and Roberto Basili. 2013. Unitor-HMM-TK: Structured kernel-based learning for spatial role labeling. In *Proceedings of SemEval 2013*, pages 573–579.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, pages 449–454.

Thorsten Joachims. 1999. Making large-scale SVM learning practical. In *Advances in Kernel Methods - Support Vector Learning*, pages 44–56. MIT Press.

Karin Kipper, Hoa Trang Dang, and Martha Palmer. 2000. Class-based construction of a verb lexicon. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and the Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 691–696.

Oleksandr Kolomiyets, Parisa Kordjamshidi, Steven Bethard, and Marie-Francine Moens. 2013. SemEval-2013 Task 3: Spatial role labeling. In *Proceedings of SemEval 2013*, pages 255–266.

Parisa Kordjamshidi and Marie-Francine Moens. 2014. Global machine learning for spatial ontology population. *Web Semantics: Science, Services and Agents on the World Wide Web*.

Parisa Kordjamshidi, Marie-Francine Moens, and Martijn van Otterlo. 2010. Spatial role labeling: Task definition and annotation scheme. In *Proceedings of 7th International Conference on Language Resources and Evaluation*, pages 413–420.

Parisa Kordjamshidi, Martijn Van Otterlo, and Marie-Francine Moens. 2011. Spatial role labeling: Towards extraction of spatial relations from natural language. *ACM Transactions on Speech and Language Processing*, 8(3):4.

Parisa Kordjamshidi, Steven Bethard, and Marie-Francine Moens. 2012. SemEval-2012 task 3: Spatial

role labeling. In *Proceedings of SemEval 2012*, pages 365–373.

Inderjeet Mani, Christy Doran, Dave Harris, Janet Hitzeman, Rob Quimby, Justin Richer, Ben Wellner, Scott Mardis, and Seamus Clancy. 2010. SpatialML: annotation scheme, resources, and evaluation. *Language Resources and Evaluation*, 44(3):263–280.

George A. Miller. 1995. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

James Pustejovsky, Jessica Moszkowicz, and Marc Verhagen. 2013. A linguistically grounded annotation language for spatial information. *ATALA: Association pour la Traitment Automatique des Langues*, 53(2):87–113.

Kirk Roberts and Sanda M. Harabagiu. 2012. UTD-SpRL: A joint approach to spatial role labeling. In *Proceedings of SemEval 2012*, pages 419–424.

Kirk Roberts, Michael A. Skinner, and Sanda M. Harabagiu. 2013. Recognizing spatial containment relations between event mentions. In *Proceedings of the 10th International Conference on Computational Semantics*.

Philip J. Stone, Dexter C. Dunphy, and Marshall S. Smith. 1966. *General Inquirer: A Computer Approach to Content Analysis*. The MIT Press.