

Mining Possessions: Existence, Type and Temporal Anchors

Dhivya Chinnappa and Eduardo Blanco

Human Intelligence and Language Technologies Lab

University of North Texas

Denton, TX, 76203

dhivyainfantchinnappa@my.unt.edu, eduardo.blanco@unt.edu

Abstract

This paper presents a corpus and experiments to mine possession relations from text. Specifically, we target alienable and control possessions, and assign temporal anchors indicating when the possession holds between possessor and possessee. We present new annotations for this task, and experimental results using both traditional classifiers and neural networks. Results show that the three subtasks (predicting possession existence, possession type and temporal anchors) can be automated.

1 Introduction

Every language has a way of expressing possessive relationships (Aikhenvald and Dixon, 2012). Possession is an asymmetric semantic relation between two entities, where one entity (the possessee) *belongs* to the other entity (the possessor) (Stassen, 2009). When it comes to defining possession, *belongs* includes a wide range of relationships, including (hereafter, we use x to refer to the possessor, and y to refer to the possessee) kinship (e.g., $[my]_x$ *oldest* $[son]_y$), part-whole (e.g., $the [car]_x$ ’s $[dashboard]_y$), physical and temporary possession (e.g., $[I]_x$ *have* $John$ ’s $[book]_y$), possession of something intangible (e.g., $[John]_x$ *got the* $[flu]_y$ *last year*) and proximity (e.g., $The [shelf]_x$ *has a* $[glass sculpture]_y$).

Possession relations can be divided into alienable (also referred to as acquired, transferable, non-intimate, etc.) and inalienable (also referred to as inherent, inseparable, intimate, etc.). Possesseees that can be separated from their possessors are alienable, and possesseees that cannot normally be separated from their possessors are inalienable (Heine, 1997). For example, $[John]_x$ ’s $[condo]_y$ is alienable, and $[John]_x$ ’s $[arm]_y$ is inalienable (some previous works would call the latter a part-whole relation instead). Tham (2004)

defines control possession as a relation in which the possessor has temporary control of the possessee, but does not necessarily alienably possess it (e.g., $[John]_x$ *borrowed the* $[car]_y$ *for the week-end*). Following the aforementioned works, possession goes beyond ownership of property.

Possession relations can be expressed in a wide variety of syntactic constructions, including noun phrases (e.g., $[John]_x$ ’s $[car]_y$) and clauses (e.g., $[John]_x$ *bought a* $[blue car]_y$). The subject of a verb can map to either the possessor as exemplified above, or to the possessee (e.g., *The* $[car]_y$ *belongs to* $[John]_x$) (Aikhenvald and Dixon, 2012).

Within computational linguistics, possession relationships have usually been studied as part of larger studies that target all relations between arguments connected with a syntactic pattern (e.g., possessive constructions, nominals). Additionally, previous efforts have mostly targeted alienable possession—or alternatively, ownership. The work presented here takes a different approach. We start by pairing people (plausible possessors) with physical objects (plausible possesseees). Then, we determine whether a possession relationship exists, and if so, (a) determine the type (alienable or control) and (b) assign temporal anchors with respect to the event of which the possessor is the subject. We target all verbs, not only prototypical verbs of possession (e.g., *have*, *get*). Thus, our approach extracts possessions intuitive to humans when there is no specific possession cue (e.g., we extract a control possession from *The* $[computer]_y$ *at work was slow*, $[I]_x$ *didn’t get anything done*).

The main contributions of this paper are: (a) deterministic procedure to pair plausible possessors and possesseees; (b) corpus annotating possession existence, possession type and temporal anchors; (c) detailed corpus analysis per verb and type of possession; and (d) experimental results showing that the task can be automated.

2 Possession Relations

The literature has studied possession relations extensively from theoretical and conceptual points of views. Here, we succinctly present some of the most influential works in the area.

The very definition of possession is not set in stone. Aikhenvald (2013) distinguishes three core meanings for possessive noun phrases that occur across languages: ownership (of property), whole-part (often referred to as part-whole), and kinship. Following a cross-linguistic perspective, she discusses possessions and time (present and former possession relationships, e.g., *my tooth* vs. *my former axe*), temporary and permanent possession (e.g., *borrow* vs. *acquire*) and others. Heine (1997) classifies possession relationships depending on the possessor and possessee. First, he makes a distinction between human (e.g., $[I]_x$ *have* $[house]_y$) and non-human possessors (e.g., $[This\ house]_x$ *has* $[two\ bedrooms]_y$). Second, he differentiates three kinds of possession depending on the possessee: concrete possession (e.g., $[I]_x$ *have* $[two\ cats]_y$), social possession (e.g., $[I]_x$ *have* $[two\ sisters]_y$), and abstract possession (e.g., $[I]_x$ *have* $[an\ idea]_y$). Miller and Johnson-Laird (1976) differentiate between three kinds of possession: inherent, accidental, and physical; and provide the following example: *He owns an umbrella (inherent), but she's borrowed it (accidental), though she doesn't have it with her (physical)*.

Possession relationships have also been defined in terms of their parameters. Stassen (2009) consider two parameters: permanent contact and control. These parameters are binary, and four kinds of possessions emerge from combining them: alienable (permanent contact: +, control: +), inalienable (+, -), temporary (-, +), and abstract (-, -). Similarly, Heine (1997) defines five binary parameters: human possessor, concrete possessee, spatial proximity, temporal permanence, and control. Combining these parameters, he defines 7 kinds of possessions: alienable, physical, temporary, inalienable, abstract, inanimate inalienable and inanimate alienable possession.

Most influential to the work presented here, Tham (2004) presents four types of possession: (a) inalienable (e.g., *John has a daughter*), (b) alienable (e.g., *John has a car*), (c) control (e.g., *John has the car (for the weekend)*), and (d) focus (e.g., *John has the window (to clean)*). In this paper, we target alienable and control posses-

sions. We discard *inalienable* possessions because automated extraction has been studied before—at least partially, e.g., part-whole (Girju et al., 2006)—and *focus* possessions because they only occurred 5 times in the corpus we work with.

3 Previous Work

Within computational linguistics, possession relations have been mostly studied as one of the many relations encoded in a given syntactic construction. For example, Tratz and Hovy (2013) extract semantic relations within English possessives. They propose a set of 18 relations, e.g. *temporal* (e.g., $[today]_x$'s $[rates]_y$), *extent* (e.g., $[6\ hours]_y$ ' $[drive]_x$). Their *controller / owner / user* relation (one relation with three aliases) is the closest relation to the alienable and control possessions we target in this paper. Unlike them, we distinguish between alienable and control possessions, and assign temporal anchors to possessions. Additionally, we are not restricted to possessive constructions. Instead, we start by pairing potential possessors and possessee within a sentence.

Extracting semantic relations between noun compounds (Nakov and Hearst, 2013; Tratz and Hovy, 2010) usually includes extracting possession relations, e.g., $[family]_x$ $[estate]_y$. Because they target noun compounds, they disregard numerous possessions encoded in text at the clause or sentence level. Although they do extract many relations from noun compounds beyond possessions, they do not distinguish between alienable and control possessions, or temporally anchor relations with respect to events in which the possessor participates.

To the best of our knowledge, the work by Banea et al. (2016) is the only one on extracting possession relations without imposing syntactic constraints. They build a dataset working with blog texts, but do not present results on automatic extraction. Their definition of possession includes alienable and control possessions, but they do not distinguish between them. Additionally, they only consider as possessors the author of a blog, and as possessee concrete nouns in the blog posts by the possessor. Regarding time, they annotate possessions at the time of the utterance. Unlike them, we distinguish between alienable and control possessions, and assign temporal anchors with respect to an event in which the possessor participates.

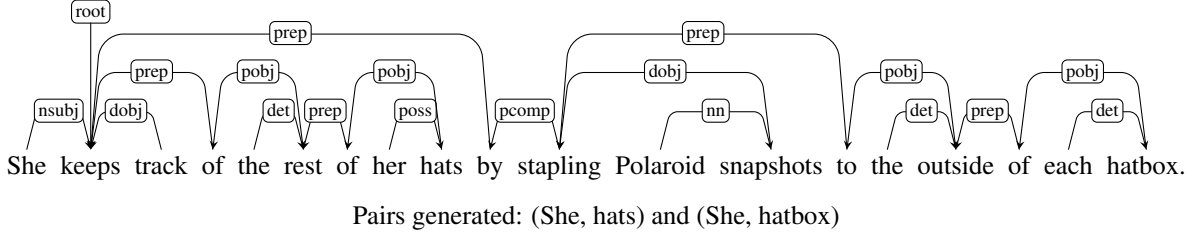


Figure 2: Sample sentence, dependency tree and pairs (x, y) generated with the steps in Section 4.1.

antiquity.n.01, block.n.01, cone.n.01, container.n.01, covering.n.02, decker.n.01, device.n.01, fabric.n.01, fixture.n.01, float.n.01, furnishing.n.01, insert.n.01, layer.n.01, lemon.n.01, marker.n.01, plaything.n.01, ready-made.n.01, squeaker.n.01, strip.n.01, vehicle.n.01
--

Figure 1: WordNet synsets used to restrict possesseees (y) when generating pairs (x, y) . *lemma.n.y* indicates the y th synset of noun *lemma*.

4 A Corpus of Possession Relations

We create a corpus¹ following two steps. First, we generate intrasentential pairs (x, y) of potential possessors (x) and possesseees (y). Second, we annotate whether a possession exists, and if so, the type and temporal anchors. Generating pairs a priori proved more effective than giving annotators plain text and asking them to annotate possessions.

We add our annotations to OntoNotes (Hovy et al., 2006). Doing so has several advantages. First, OntoNotes contains texts from several domains and genres (e.g., conversational telephone speech, weblogs, broadcast), thus we not only work with newswire. Second, OntoNotes includes part-of-speech tags, named entities and parse trees, three annotation layers that allow us to streamline the corpus creation process.

4.1 Pairing Potential Possessors and Possesseees

Our goal is to obtain pairs (x, y) such that it is plausible that x is the possessor of possessee y . To do so, we follow these steps:

1. Collect as potential possessors all PERSON named entities and personal pronouns (part-of-speech tag PRP) *I, he, she, we* and *they*.
2. Discard potential possessors that are not the nominal subject (*nsubj* syntactic dependency) of a verb. Let us name that verb *verb_x*.

possessee (y) is subsumed by	possessor (x) is a		
	pronoun	person NE	All
device.n.01	295	74	369
container.n.01	189	30	219
covering.n.02	138	54	192
vehicle.n.01	124	36	160
fabric.n.01	6	7	13
block.n.01	9	2	11
plaything.n.01	3	2	5
fixture.n.01	4	0	4
antiquity.n.01	2	0	2
other	4	0	4
All	774	205	979

Table 1: Counts of pairs (x, y) generated per type of potential possessor (x) and possessee (c).

3. For each possessor, collect as potential possesseees all nouns reachable from *verb_x* in the dependency tree and subsumed in WordNet (Miller, 1995) by the synsets in Figure 1.

Step (1) selects most people (not groups), and is inspired by Aikhenvald (2013, p. 11), who states that possessors are usually animate. Step (2) reduces the number of potential possessors, but note that we do not impose any restriction on *verb_x*, which may or may not be a verb of caused possession (Beavers, 2011). Finally, Step (3) restricts the kind of objects considered as possesseees. The list of synsets was defined after analyzing the WordNet noun hierarchy and prior to generating pairs. Most of these synsets are children of *artifact.n.01*, other children of *artifact.n.01* were discarded because intuitively they cannot be possesseees. For example, we discard *mystification.n.02*: *something designed to mystify or bewilder*.

Figure 2 shows a sample sentence and the pairs generated. Note that these pairs include distant possessor-possessee pairs, not only subject-object pairs. Nouns *track*, *rest*, *Polaroid* and *snapshot* are discarded as potential possesseees because they are not subsumed by the synsets in Figure 1.

¹Available at <http://www.cse.unt.edu/~blanco>

Labels	%	κ
yes, never, unk, inv	86.1	0.79
alienable, control	82.5	0.77
before yes, before no	83.6	0.68
during yes, during no	88.8	0.75
after yes, after no	83.6	0.59

Table 2: Inter-annotator agreements (raw percentage and Cohen’s κ). κ values in the 0.60–0.80 range are considered *substantial*, over 0.80 would be *perfect* (Artstein and Poesio, 2008).

The total number of pairs generated after executing Steps (1–3) is 2,025. In order to reduce the annotation effort, we set to annotate 1,000 pairs. After trying several strategies, we reduce the number of pairs as follows. First, we discard pairs with verb_x *see*, *think*, *believe*, *say* and *tell* because pilot annotations revealed that almost no possessions can be extracted from them (1,757 pairs left). Second, we discard pairs (x, y) such that verb_x occurs five or less times (979 pairs left). Table 1 presents basic counts per type of possessor (named entity or personal pronoun) and possessee (WordNet synset) for the 979 pairs.

4.2 Annotating Possession Existence, Types and Temporal Information

After automatically generating pairs of potential possessors and possesseees, annotators validate them manually. Annotations were done in-house, and the annotation interface showed the current sentence (with x , y and verb_x highlighted), as well as the previous and next sentences.

The annotation process includes two major steps. First, annotators decide whether a possession relation exists between x and y based on the three sentences provided. More specifically, they choose from the following labels:

- *yes* if a possession exists at some point of time with respect to verb_x ;
- *never* if a possession does not exist at any point of time with respect to verb_x ;
- *unk* if it is sound to ask whether x is the possessor of possessee y , but there is not enough information to choose *yes* or *never*; and
- *inv* if either the potential possessor x is not animate, or the potential possessee y is non-sensical in the given context.

Second, annotators make two more decisions if the first label is *yes*:

- Possession type: whether the possession is *alienable* or *control*.

	Before		During		After	
	<i>yes</i>	<i>no</i>	<i>yes</i>	<i>no</i>	<i>yes</i>	<i>no</i>
Alienable	69.8	30.2	55.9	44.1	92.6	7.4
Control	28.8	71.2	85.3	14.7	33.3	66.7
All	52.0	48.0	68.7	31.3	67.8	32.2

Table 3: Percentage of alienable and control POSSESSION relations annotated *yes* and *no* per temporal anchor (before, during, after) with respect to verb_x (i.e., the verb of which the possessor is the subject).

- Temporal anchors: whether the possession is true at some point of time *before*, *during*, and at some point of time *after* verb_x takes place (three binary decisions).

Following the literature (Tham, 2004), we define *alienable* possession as a possessor owning a possessee, and *control* possession as a possessor having control of the possessee, but not necessarily ownership. Annotators were instructed to use world knowledge and fully interpret the sentences provided beyond what is explicitly stated. We present annotation examples in Section 5.1

Inter-Annotator Agreement. The annotations were done by two graduate students. Both of them annotated 35% of all pairs (possession existence, possession type and temporal anchors). We show inter-annotator agreements in Table 2. Cohen’s κ for possession detection (labels *yes*, *never*, *unk* and *inv*) is 0.79, and 0.77 when including possession type (labels *alienable* and *control*). Answering whether the possession is true before, during or after verb_x obtains lower coefficients: 0.68, 0.75 and 0.59 respectively. Not surprisingly, the agreement for *during* is higher. Note that κ coefficients in the range 0.60–0.80 are considered *substantial*, and coefficients over 0.80 are usually considered *perfect* (Artstein and Poesio, 2008). Given these high agreement, the rest of pairs (65%) were annotated once.

5 Corpus Analysis

Figure 3 presents percentages per label for all verbs and the top 10 most frequent verbs. Overall, 36.5% of pairs are validated (*alienable*: 20.6%, *control*: 15.9%), and only 5.8% of pairs are annotated *unk*. The relatively high percentage of *inv* label is mostly due to potential possesseees that can only be possessed in certain contexts, e.g., compare $[They]_x$ *asked* $[regulators]_y$ *to suggest new ways to [...]* (*inv*) vs. $[They]_x$ *replaced the* $[regulators]_y$ *to control the flow of water* (*yes*).

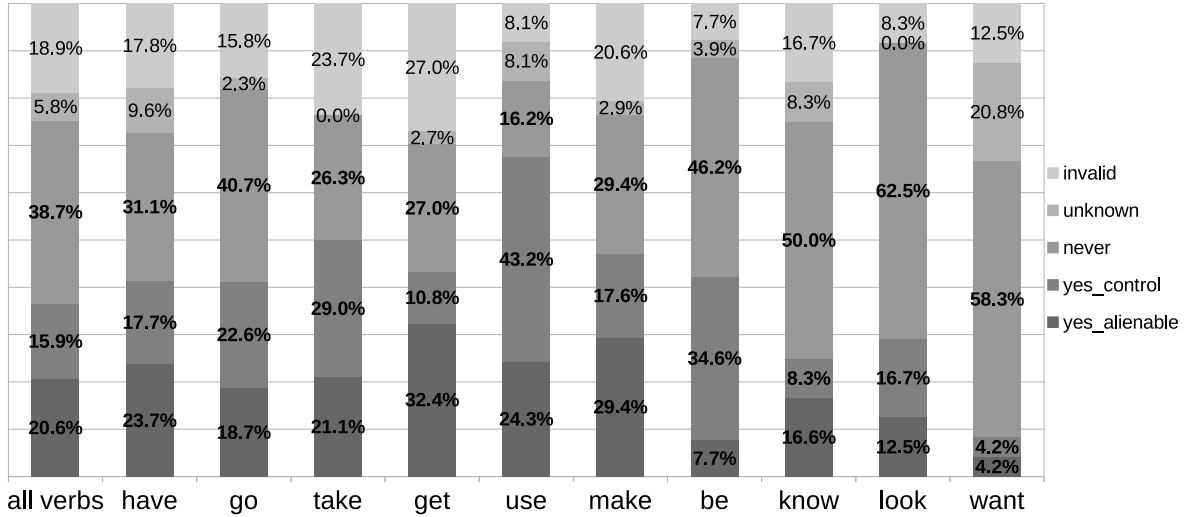


Figure 3: Percentage of labels for all verbs and the top 10 most frequent verbs of which the possessor is the subject. From left to right, they occurred 135, 44, 38, 37, 37, 34, 26, 24, 24, and 24 times respectively.

The percentage distributions depends heavily on the verb at hand. Note that several verbs with high *alienable* and *control* labels are not prototypical verbs of possession (e.g., *go*, *use*, *know*). When a possession holds, the type is most likely *control* for most verbs. The only exceptions are *have* (23.7% vs. 17.7%), *get* (32.4% vs. 10.8%), *make* (29.4% vs. 17.6%) and *know* (16.6% vs. 8.3%). The most productive verb as far as alienable possession is *get* (32.4%), and as far as *control* possessions, *use* (43.2%).

Labels per temporal anchor with respect to verb_x (binary flags for before, during and after) and possession type are presented in Table 3. Alienable and control possessions show opposite trends for *before* and *after*, and substantially different distributions for *during*. The vast majority of control possessions are true during verb_x (85.3% vs. 14.7%), as well as a more modest majority of alienable possessions (55.9% vs. 44.1%). Alienable and control possessions, however, have opposite temporal anchors for *before* and *after*. Specifically, most alienable possessions are true before and after verb_x (69.8% and 92.6% respectively), and most control possessions are not true before and after verb_x (71.2% and 66.7%).

5.1 Examples of Annotations

We present annotation examples using selected pairs of possessors and possesseees in Table 4.

In Sentence (1), annotators interpreted that the relationship between *he* and *car* is an alienable possession. While not explicitly stated, annotators interpreted that *he* is an adult, and world knowl-

edge tells us that most adults own the cars they drive unless a modifier indicates otherwise (e.g., *rental car*, *my father's car*). Regarding temporal anchors, the possession between *he* and *car* is true before and during *died*, but not after.

Sentence (2) is a common example of alienable possession that is true after verb_x . The subject of a verb of creation (e.g., *make*, *build*) often becomes an alienable possessor of the direct object after the verb, but not before or during (because the object has not come into being yet).

Sentence (3) and (4) exemplify control possessions. In Sentence (3), *He* is borrowing *my father's car* for a period of time, and thus *He* has control over but does not own it. Regarding temporal anchors, nothing in the sentence indicates that *He* will have control over the car before or after *kept*. Note that our procedure to generate pairs would not generate the pair (*father*, *car*), but previous work has targeted possessives (Section 3).

In Example (4), verb_x is *felt*, yet we extract a valid control possession. *I* is crew member of a warship and is describing his experience while on board. Annotators understood he had control over the ship (at least partially) before, during and after, as *felt* did not last long and there is no indication that *I* left the boat immediately before or after *felt*.

Sentences (5–7) present examples in which annotators did not annotate a possession relation (labels *never*, *unk*, and *inv*). In Sentence (5), the *mask* belongs to *Joseph*. There is no indication that a possession relation exists between *LaToya* and *mask*, although *LaToya* was in close spatial proximity of the *mask* worn by *Joseph*.

	Text (sentence of interest and context if relevant) and pair (x, y)	Label	B	D	A
1	But [he] _x had extreme mood swings and [died] _{verb_x} in a [car] _y crash driving to work when I was five.	alienable	✓	✓	✗
2	After moving to the unoccupied zone, [Wang] _x began [carving] _{verb_x} [seals] _y in his spare time to support himself.	alienable	✗	✗	✓
3	[He] _y [kept] _{verb_x} my father’s [car] _y for a year, without writing a confiscation order for it.	control	✗	✓	✗
4	The horror and the heroics of the [...] were relived this evening by one of the crew, Lieutenant Ann Chamberlain. [I] _x just [felt] _{verb_x} a large—not so much a bang, but, to me, from where I was, it seemed like something had rammed the [ship] _y .	control	✓	✓	✓
5	[LaToya] _x also [described] _{verb_x} being awakened in the night by Joseph wearing a “monster [mask] _y .”	never	n/a		
6	[They] _x [asked] _{verb_x} him to come to them immediately because the reported [car] _y had been seized.	unk	n/a		
7	Will your political party straighten up and say, damn it, [we] _x [have] _{verb_x} to drop some of our ideological [baggage] _y ?	inv	n/a		

Table 4: Example of annotations for selected pairs (x, y). Recall that temporal anchors (B: before, D: during, A: After) are annotated with respect to verb_x and only if the main label is yes, i.e., if there is an alienable or control POSSESSION between x and y (Section 4.2).

In Sentence (6), it is the case that *They* have some knowledge about the *car* that was seized, and it appears that *him*—not *They*—may be the alienable possessor. It is unclear, however, whether *They* and *car* are related by a control possession, thus annotators chose label *unk*.

Finally, Sentence (7) exemplifies label *inv*. While *baggage* is most of the time a concrete object that passes the restrictions on potential possessors (Section 4), in this context, it is part of the metaphor *ideological baggage*. Since we only target concrete possessors, annotators chose *inv*.

6 Experiments and Results

We conduct experiments using Support Vector Machines and neural networks. Each pair (x, y) becomes an instance, and we create stratified train (80%) and test (20%) sets. We report results using the test set after tuning hyper parameters using 10-fold cross validation. More specifically, we train five classifiers and experiment with all instances but the ones annotated *inv*. The first classifier predicts possession existence (yes, never or unk). The second classifier predicts possession types, i.e., classifies pairs between which a possession holds (yes) into *alienable* or *control*. The third, fourth and fifth classifiers predict temporal anchors, i.e., classify pairs between which a possession holds—either *alienable* or *control*—into *before yes* or *before no*, *during yes* or *during no*, and *after yes* or *after no*.

6.1 Support Vector Machines

We trained the five classifiers using the SVM implementation in scikit-learn (Pedregosa et al., 2011). We tuned hyper-parameters *C* and *γ* using 10-fold cross validation, and used the features that are summarized in Table 5.

Verb features include the word and POS tag for the verb, previous and next tokens, as well as information regarding the outgoing and incoming dependencies. We also include a binary flag indicating whether the verb is a possession verb from the list collected by Viberg (2010, Table 1).

Possessor and *Possessee* features are very similar to *Verb* features, but we consider the concatenation of words and POS tags. *Possessee* features also include information derived from the WordNet hypernym paths to the root in the noun hierarchy, i.e., *entity.n.01*. More specifically, *WN synset* captures the synset from Figure 1 the possessee is subsumed by, and *WN path* are features capturing the top 6 synsets in the hypernym path from the possessee to *entity.n.01*. Finally, *Path* features include three syntactic paths (syntactic dependency types and up / down symbols): from the possessor to the verb, from the possessee to the verb, and from the possessor to the possessee. The feature set is heavily inspired in many previous works (e.g., (Gildea and Jurafsky, 2002)).

We experimented with SVMs to establish a strong supervised baseline using linguistic information, and to compare with neural networks that take as input only words along with information

	Feature	Description
Verb	word and tag	word form and part-of-speech tag of verb
	is_possession_verb	flag indicating whether <i>verb</i> is in the list of possession verbs
	previous, next tokens	word form and part-of-speech tags of the previous and next tokens
	dependency_out	outgoing syntactic dependency type
	dependencies_in	flags indicating the incoming syntactic dependencies
	left, right children	number of incoming syntactic dependencies to the left and right of <i>verb</i>
Possessor	words	concatenation of words
	pos tags	part-of-speech tag (full tag and first character, i.e., pronoun or noun)
	previous, next tokens	word form and part-of-speech tags of the previous and next tokens
	dependency_out	outgoing syntactic dependency type
	dependencies_in	flags indicating the incoming syntactic dependencies
Possessee	same as possessor	same features extracted for the possessor
	WN synset	WordNet synset from Figure 1 the possessee is subsumed by
	WN path	WordNet synsets from entity.n.01 to the possessee
Paths	possessor to verb	syntactic path between possessor and verb
	possessee to verb	syntactic path between possessee and verb
	possessor to possessee	syntactic path between possessor and possessee

Table 5: Feature set used to extract possession relations (existence, type and temporal anchors) with Support Vector Machines.) and possession type (*alienable* or *control*).

regarding who is the potential possessor, possessee and verb_x .

6.2 Neural Networks

We experiment with feedforward and Long Short-Term Memory networks, and use the implementations in Keras (Chollet et al., 2015) using TensorFlow backend (Abadi et al., 2015). All networks use GloVe embeddings with 100 dimensions (Pennington et al., 2014) and the Adam optimizer (Kingma and Ba, 2014). Regarding input, we experiment with the potential possessor x , possessee y , verb_x , and the rest of the sentence. The three architectures are depicted in Figure 4.

Feedforward Neural Network. The feedforward neural network takes as input the embeddings of the potential possessor x , possessee y and verb_x . It has a fully connected hidden layer with 50 neurons and uses softmax in the output layer of size 3 for predicting possession existence (*yes*, *never* and *unk*) or size 2 for predicting possession type (*alienable* and *control*) and temporal anchors (*yes* and *never* for *before*, *during* and *after*).

LSTM_{ppv}. The first Long Short-Term Memory network takes as input a fixed-length sequence consisting of the potential possessor x , possessee y and verb_x . We used 100 LSTM units (output dimension) and the output layer also uses softmax. While this LSTM has access to the same information than the feedforward network, we expect that the input, output and forget gates will learn to update the cell state to better solve our task.

LSTM_{sent}. The architecture of the second Long Short-Term Memory network is the same than LSTM_{ppv}, but the input is different. LSTM_{sent} takes as input the sequence of words from which the potential possessor x , possessee y and verb_x were extracted. Each element in the input is represented by the concatenation of its word embedding and an additional embedding indicating if the token is the potential possessor x , possessee y , verb_x , or none of them. Unlike the other two networks, LSTM_{sent} has access to the full sentence, and we expect that the memory update mechanism (i.e., the input, output and forget gates) will learn the context most relevant for our task.

6.3 Results

Possession Existence and Type. Table 6 presents results obtained with the majority baseline (possession existence: always *never*, possession type: always *alienable*), SVMs and the three neural networks. All models outperform the majority baseline in both tasks (possession existence F1: 0.24, possession type F1: 0.40), and the three neural architectures outperform SVM (existence: 0.57–0.74 vs. 0.56, type: 0.61–0.67 vs. 0.58).

Regarding possession existence, the vanilla feedforward neural network alone performs similar to the SVM (F1: 0.57 vs. 0.56), indicating that word embeddings capture the kind of verbs and (potential) possessors and possessee more likely to have a possession relationship. Despite the small dataset (total: 979 pairs), the LSTMs out-

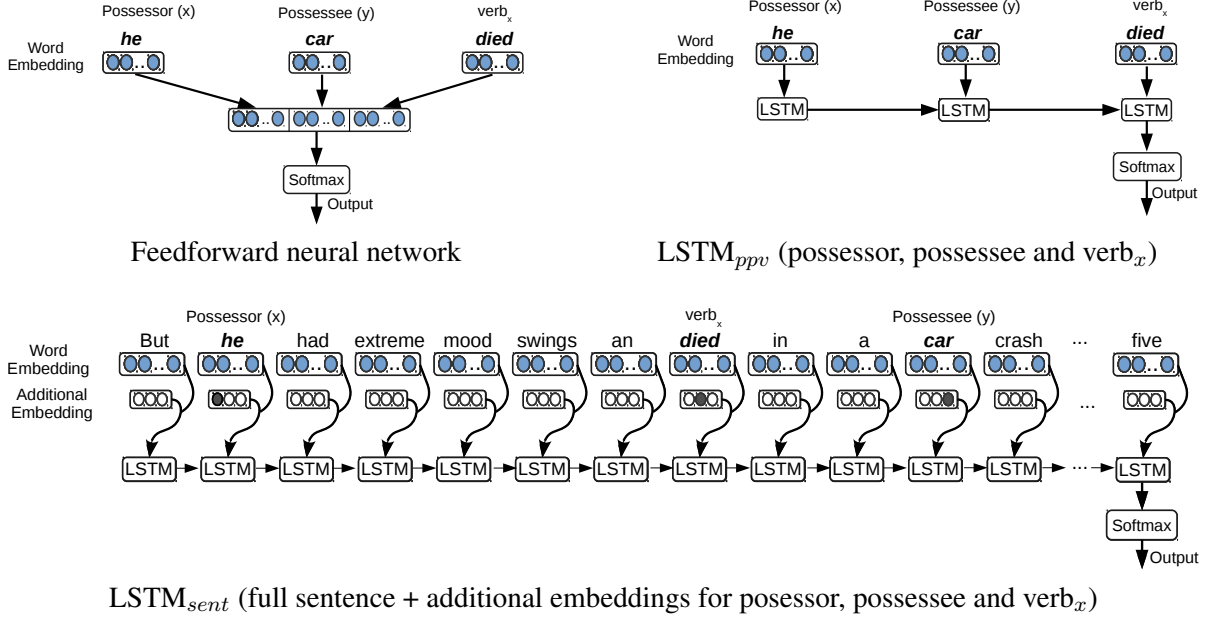


Figure 4: Neural network architectures: feedforward neural network (top left), LSTM_{ppv} (top right), and LSTM_{sent} (bottom). We exemplify the architectures with Example 1 from Table 4, *But [he]_x had extreme mood swings and [died]_{verb_x} in a [car]_y crash driving to work when I was five* (only partially shown with LSTM_{sent}).

	Maj. baseline			SVM, all feats.			FFNN			LSTM _{ppv}			LSTM _{sent}		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
yes	.00	.00	.00	.57	.65	.61	.54	.53	.54	.65	.69	.67	.74	.72	.73
never	.41	1.0	.58	.61	.53	.56	.59	.59	.59	.71	.79	.75	.73	.82	.77
unk	.00	.00	.00	.44	.43	.44	.56	.59	.58	.75	.49	.59	.80	.65	.72
Weighted Avg.	.17	.41	.24	.56	.56	.56	.57	.57	.57	.70	.69	.69	.75	.75	.74
alienable	.56	1.0	.72	.64	.57	.61	.68	.75	.71	.67	.80	.73	.67	.60	.63
control	.00	.00	.00	.53	.59	.56	.64	.56	.60	.67	.50	.57	.56	.62	.59
Weighted Avg.	.32	.56	.40	.59	.58	.58	.66	.67	.66	.67	.67	.67	.62	.61	.61

Table 6: Results obtained using the majority baseline (possession existence: *never*, possession type: *alienable*), SVMs with the best feature combination (all features), and neural networks. Note that we report results for the possession existence (*yes*, *never* or *unk*) and possession type (*alienable* or *control*).

perform the feedforward neural network (0.74 vs. 0.57). LSTM_{ppv} performs surprisingly well (F1: 0.69) even though it only has access to the possessor, possessee and verb_x. LSTM_{sent} highly benefits from having access to the full sentence (F1: 0.74). This shows that context plays a vital role in deciding the existence of possession.

Regarding possession type, the feedforward neural network is comparable to LSTM_{ppv}. Intuitively, distinguishing between alienable and control possessions can be done mostly based on the possessor, possessee and verb_x, and the embeddings capture this kind of information. For example, verbs such as *use* and *rent* indicate a control possession, while *acquire* indicates alienable possession.

Temporal Anchors. Table 7 presents results obtained with SVMs and the best neural network architecture in this subtask. LSTM_{ppv} performs similar to the SVM (before: 0.71 vs. 0.76, during: 0.75 vs. 0.72, after: 0.70 vs. 0.73). As expected, F1 scores are higher with the labels that occur more often: *yes* is more frequent than *never* with all temporal anchors, especially *during* and *after* (Table 3), and F1 scores for *yes* are higher than for *never* (before: 0.73 vs. 0.68, during: 0.82 vs. 0.59, after: 0.77 vs. 0.54).

7 Conclusions

Possession relations are present in all languages, and they can reflect relationships, values, concepts and cultural changes (Aikhenvald, 2013). In this

		Before			During			After		
		P	R	F1	P	R	F1	P	R	F1
SVM, all feats.	Yes	0.76	0.82	0.78	0.78	0.82	0.80	0.78	0.86	0.82
	No	0.77	0.71	0.74	0.57	0.52	0.55	0.61	0.48	0.54
	Weighted Avg.	0.76	0.76	0.76	0.72	0.72	0.72	0.72	0.74	0.73
LSTM _{ppv}	Yes	0.71	0.76	0.73	0.80	0.84	0.82	0.79	0.76	0.77
	No	0.71	0.65	0.68	0.62	0.57	0.59	0.52	0.57	0.54
	Weighted Avg.	0.71	0.71	0.71	0.74	0.75	0.75	0.70	0.69	0.70

Table 7: Results obtained using SVMs with the best feature combination (all features) and the best neural network architecture when predicting temporal anchors with respect to verb_y for a POSSESSION (both alienable and control).

paper, we mine possessions from text. Specifically, we extract alienable and control possessions, and specify temporal anchors with respect to the verb of which the possessor is the subject.

We have created the first corpus annotating types of possessions following two steps. First, we automatically pair potential possessors and possesseees, resulting in 979 pairs. Second, we manually validate pairs by annotating possession existence (yes, never, unk and inv), types (alienable or control) and temporal anchors (before yes / no, during yes / no, after yes / no). Inter-annotator Cohen’s κ coefficients show that the annotation task can be done reliably (Table 2). Experimental results show that the task can be automated, and that neural networks outperform SVMs trained with features extracted from linguistic structure although we experiment with a relatively small dataset.

Beyond fundamental research, we believe that mining possession types has several applications. For example, marketers may target people who do not alienably possess something, and certain skills may be inferred from the kind of objects people have control possessions over (e.g., an individual having a control possession of an 18-wheeler most likely knows how to drive large trucks and has a commercial driver’s license).

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Tal-

war, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. [TensorFlow: Large-scale machine learning on heterogeneous systems](https://www.tensorflow.org/). Software available from tensorflow.org. <https://www.tensorflow.org/>.

A.Y. Aikhenvald. 2013. Possession and ownership: a cross-linguistic perspective. In A.Y. Aikhenvald and R.M.W. Dixon, editors, *Possession and Ownership: A Cross-Linguistic Typology*, Oxford University Press, Oxford, chapter 1, pages 1–64.

A.Y. Aikhenvald and R.M.W. Dixon. 2012. *Possession and Ownership: A Cross-Linguistic Typology*. Explorations in Linguistic Typology. OUP Oxford.

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Comput. Linguist.* 34(4):555–596.

Carmen Banea, Xi Chen, and Rada Mihalcea. 2016. Building a dataset for possessions identification in text. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odiijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA), Paris, France.

J. Beavers. 2011. An aspectual analysis of ditransitive verbs of caused possession in english. *Journal of Semantics* 28(1):1–54.

François Chollet et al. 2015. Keras. <https://github.com/fchollet/keras>.

Daniel Gildea and Daniel Jurafsky. 2002. [Automatic labeling of semantic roles](https://doi.org/10.1162/089120102760275983). *Comput. Linguist.* 28(3):245–288. <https://doi.org/10.1162/089120102760275983>.

Roxana Girju, Adriana Badulescu, and Dan Moldovan. 2006. [Automatic discovery of part-whole relations](https://doi.org/10.1162/coli.2006.32.1.83). *Comput. Linguist.* 32(1):83–135. <https://doi.org/10.1162/coli.2006.32.1.83>.

B. Heine. 1997. *Possession: Cognitive Sources, Forces, and Grammaticalization*. Cambridge Studies in Linguistics. Cambridge University Press.

- Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. *OntoNotes: the 90% Solution*. In *NAACL '06: Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers on XX*. Association for Computational Linguistics, Morristown, NJ, USA, pages 57–60. <http://portal.acm.org/citation.cfm?id=1614064>.
- Diederik P. Kingma and Jimmy Ba. 2014. *Adam: A method for stochastic optimization*. *CoRR* abs/1412.6980. <http://arxiv.org/abs/1412.6980>.
- G.A. Miller and P.N. Johnson-Laird. 1976. *Language and perception*. Belknap Press. Belknap Press of Harvard University Press.
- George A. Miller. 1995. Wordnet: A lexical database for english. In *Communications of the ACM*. volume 38, pages 39–41.
- Preslav I. Nakov and Marti A. Hearst. 2013. Semantic interpretation of noun compounds using verbal and other paraphrases. *ACM Trans. Speech Lang. Process.* 10(3):13:1–13:51.
- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12:2825–2830.
- Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. *Glove: Global vectors for word representation*. In *Empirical Methods in Natural Language Processing (EMNLP)*. pages 1532–1543. <http://www.aclweb.org/anthology/D14-1162>.
- L. Stassen. 2009. *Predicative Possession*. Oxford Studies in Typology and Linguistic Theory. OUP Oxford.
- Shiao Wei Tham. 2004. *Representing Possessive Predication: Semantic Dimensions and Pragmatic Bases*. Ph.D. thesis, Stanford University.
- Stephen Tratz and Eduard Hovy. 2010. A taxonomy, dataset, and classifier for automatic noun compound interpretation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Stroudsburg, PA, USA, ACL '10, pages 678–687.
- Stephen Tratz and Eduard H. Hovy. 2013. Automatic interpretation of the english possessive. In *ACL (1)*. The Association for Computer Linguistics, pages 372–381.
- Ake Viberg. 2010. Basic verbs of possession. *Cogni-Textes* 4.