Relation Extraction from the Web using Distant Supervision

Isabelle Augenstein, Diana Maynard and Fabio Ciravegna

Department of Computer Science, The University of Sheffield, UK
{i.augenstein,d.maynard,f.ciravegna}@dcs.shef.ac.uk

Abstract. Extracting information from Web pages requires the ability to work at Web scale in terms of the number of documents, the number of domains and domain complexity. Recent approaches have used existing knowledge bases to learn to extract information with promising results. In this paper we propose the use of distant supervision for relation extraction from the Web. Distant supervision is a method which uses background information from the Linking Open Data cloud to automatically label sentences with relations to create training data for relation classifiers. Although the method is promising, existing approaches are still not suitable for Web extraction as they suffer from three main issues: data sparsity, noise and lexical ambiguity. Our approach reduces the impact of data sparsity by making entity recognition tools more robust across domains, as well as extracting relations across sentence boundaries. We reduce the noise caused by lexical ambiguity by employing statistical methods to strategically select training data. Our experiments show that using a more robust entity recognition approach and expanding the scope of relation extraction results in about 8 times the number of extractions, and that strategically selecting training data can result in an error reduction of about 30%.

1 Introduction

Almost all of the big name Web companies are currently engaged in building 'knowledge graphs' and these are showing significant results in improving search, email, calendaring, etc. Even the largest openly-accessible ones, such as Freebase [4] and Wikidata [24], are however far from complete. Most of the missing information is available in the form of free text on Web pages. To access that knowledge and populate knowledge bases, text processing methods such as relation extraction are necessitated. In this paper, we understand relation extraction as the problem of extracting relations, e.g. origin(musical artist, location), for entities, e.g. "The Beatles" of certain classes (e.g. musical artist). One important aspect to every relation extraction approach is how to annotate training and test data for learning classifiers. In the past, four groups of approaches have been proposed (see also Section 2).

Supervised aproaches use manually labelled training and test data. Those approaches are often specific for, or biased towards a certain domain or type of text. This is because information extraction approaches tend to have a higher performance if training and test data is restriced to the same narrow domain. In addition, developing supervised approaches for different domains requires even more manual effort.

Unsupervised approaches do not need any annotated data for training and instead extract words between entity mentions, then cluster similar word sequences and generalise them to relations. Although unsupervised aproaches can process very large amounts of data, the resulting relations are hard to map to ontologies. In addition, it has been documented that these approaches often produce uninformative as well as incoherent extractions [7].
Semi-supervised methods only require a small number of seed instances. The handcrafted seeds are used to extract patterns from a large corpus, which are then used to extract more instances and those again to extract new patterns in an iterative way. The selection of initial seeds is very challenging - if they do not accurately reflect the knowledge contained in the corpus, the quality of extractions might be low. In addition, since many iterations are needed, these methods are prone to semantic drift, i.e. an unwanted shift of meaning. This means these methods require a certain amount of human effort - to create seeds initially and also to help keep systems "on track" to prevent them from semantic drift.
A fourth group of approaches are *distant supervision* or *self-supervised* learning approaches. The idea is to exploit large knowledge bases (such as Freebase [4]) to automatically label entities in text and use the annotated text to extract features and train a classifier. Unlike supervised systems, these approaches do not require manual effort to label data and can be applied to large corpora. Since they extract relations which are defined by vocabularies, these approaches are less likely to produce uninformative or incoherent relations.
Although promising, distant supervision approaches have so far ignored issues arising in the context of Web extraction and thus still have limitations that require further research. Note that some of those issues are not specific to distant supervision and have been researched for supervised, semi-supervised or unsupervised approaches. To illustrate those limitations, consider the following example:
"*Let It Be* is the twelfth and final album by *The Beatles* which contains their hit single '*Let it Be*'. The band broke up in 1974."
Unrecognised Entities: Distant supervision approaches use named entity classifiers that recognise entities that were trained for the news domain. When applying those approaches to heterogenous Web pages, types of entities which do not exist in that domain are not recognised. Two of those types are *MusicalArtist:track* and *MusicalArtist:album*, i.e. *Let It Be* would not be recognised.
Restrictive assumption: Existing distant supervision systems only learn to extract relations which do not cross sentences boundaries, i.e. sentences which contain an explicit mention of the name of both the subject and the object of a relation. This results in data sparsity. In the example above, the second sentence does not contain two named entities, but rather a pronoun representing an entity and a NE. While coreference resolution tools could be applied to detect the NE the pronoun refers to, those tools have a low performance on heterogeneous Web pages, where formatting is often used to convey coreferences and linguistic anomalies occur, and because they are based on recognising the NE in the first place.
Ambiguity: In the first sentence, the first mention of *Let It Be* is an example for the *MusicalArtist:album* relation, whereas the second mention is an example of the *MusicalArtist:album* relation. If both mentions are used as positive training data for both

relations, this impairs the learning of weights of the relation classifiers. This aspect has already been partly researched by existing distant supervision approaches.
Setting: The general setting of existing distant supervision approaches is to assume that every text might contain information about any possible property. Making this assumption means that the classifier has to learn to distinguish between all possible properties, which is unfeasable with a large domain and a big corpus.

This paper aims to improve the state of the art in distant supervision for Web extraction by: (1) recognising named entities across domains on heterogeneous Web pages by using Web-based heuristics; (2) to report results for extracting relations across sentence boundaries by relaxing the distant supervision assumption; (3) to propose statistical measures for increasing the precision of distantly supervised systems by filtering ambiguous training data; and (4) to document an entity-centric approach for Web relation extraction using distant supervision.

2 Related Work

There are have been several different approaches for information extraction from text for populating knowledge bases which try to minimise manual effort in the recent past.
Semi-supervised bootstrapping approaches such as NELL [5], PROSPERA [15] and BOA [9] start with a set of seed natural language patterns, then employ an iterative approach to both extract information for those patterns and learn new patterns. For NELL and PROPERA, the patterns and underlying schema are created manually, whereas they are created automically for BOA by using knowlege contained in DBpedia.
Ontology-based question answering systems often use patterns learned by semi-supervised information extraction approaches as part of their approach, Unger et al. [23], for instance, use patterns produced by BOA.
Open information extraction (Open IE) approaches such as TextRunner [27], Reverb [7], OLLIE [12] and ClausIE [6] are unsupervised approaches, which learn cross-relation extraction patterns from text. Although they can process very large amounts of data, the resulting relations are hard to map to desired ontologies or user needs, and can often produce uninformative or incoherent extractions, as mentioned in Section 1.
Bootstrapping and Open IE approaches differ from our approach in the respect that they are rule-based, not statistical approaches, i.e. they learn natural language patterns, not weights for features for a machine learning model. The difference between them is that statistical approaches take more different factors into account and make 'soft' judgements, whereas rule-based approaches make hard judgments based on prominant patterns. Rule-based approaches do not require much training data and generally have a good performance on narrow domains. However, maintaining and developing those approaches can be very time-consuming. In contrast, statistical methods are easy to develop if sufficient training data is available. Once developed, the are more robust to unseen information, i.e. if the training and test data are drawn from different domains, or if unseen words or sentence contructions occur. We opt for a statistical approach, since we aim at extracting information from heterogenous Web pages.
Automatic ontology learning and population approaches such as FRED [17] and LODifier [3] extract an ontology schema from text, map it to existing schemas and extract

information for that schema. Unlike bootstrapping approaches, they do not employ an iterative approach. However, they rely on several existing natural language processing tools trained on newswire and are thus not robust enough for Web information extraction.
Finally, *distantly supervised or semi-supervised approaches* aim at exploiting background knowledge for relation extraction, most of them for extracting relations from Wikipedia. Mintz et al. [14] describe one of the first distant supervision approaches which aims at extracting relations between entities in Wikipedia for the most frequent relations in Freebase. They report precision of about 0.68 for their highest ranked 10% of results depending what features they used. In contrast to our approach, Mintz et al. do not experiment with changing the distant supervision assumption or removing ambiguous training data, they also do not use fine-grained relations and their approach is not class-based. Nguyen et al. [16]'s approach is very similar to that of Mintz et al. [14], except that they use a different knowledge base, YAGO [20]. They use a Wikipedia-based named entity recogniser and classifier (NERC), which, like the Stanford NERC classifies entities into persons, relations and organisations. They report a precision of 0.914 for their whole test set, however, those results might be skewed by the fact that YAGO is a knowledge base derived from Wikipedia.
A few strategies for seed selection for distant supervision have already been investigated: at-least-one models [10,21,18,26,13], hierarchical topic models [1,19], pattern correlations [22], and an information retrieval approach [25]. At-least-one models [10,21,18,26,13] are based on the idea that "if two entities participate in a relation, at least one sentence that mentions these two entities might express that relation" [18]. While positive results have been reported for those models, Riedel et al. [18] argues that it is challenging to train those models because they are quite complex. Hierarchical topic models [1,19] assume that the context of a relation is either specific for the pair of entities, the relation, or neither. Min et al. [13] further propose a 4-layer hierarchical model to only learn from positive examples to address the problem of incomplete negative training data. Pattern correlations [22] are also based on the idea of examining the context of pairs of entities, but instead of using a topic model as a pre-processing step for learning extraction patterns, they first learn patterns and then use a probabilistic graphical model to group extraction patterns. Xu et al. [25] propose a two-step model based on the idea of pseudo-relevance feedback which first ranks extractions, then only uses the highest ranked ones to re-train their model.
Our research is based on a different assumption: Instead of trying to address the problem of noisy training data by using more complicated multi-stage machine learning models, we want to examine how background data can be even further exploited by testing if simple statistical methods based on data already present in the knowledge base can help to filter unreliable training data. Preliminary results for this have already been reported in Augenstein [2]. The benefit of this approach compared with other approaches is that it does not result in an increase of run-time during testing and is thus more suited towards Web-scale extraction than approaches which aim at resolving ambiguity during both training and testing. To the best of our knowledge, our approach is the first distant supervision approach to address the issue of adapting distant supervision to relation extraction from heterogeneous Web pages and to address the issue of data sparsity by relaxing the distant supervision assumption.

3 Distantly Supervised Relation Extraction

Distantly supervised relation extraction is defined as automatically labelling a corpus with properties, P and resources, R, where resources stand for entities from a knowledge base, KB to train a classifier to learn to predict binary relations. The distant supervision paradigm is defined as follows: [14]:

> If two entities participate in a relation, any sentence that contains those two entities might express that relation.

In general relations are of the form $(s, p, o) \in R \times P \times R$, consisting of a subject, a predicate and an object; during training, we only consider statements, which are contained in a knowledge base, i.e. $(s, p, o) \in KB \subset R \times P \times R$. In any single extraction we consider only those subjects in a particular class $C \subset R$, i.e. $(s, p, o) \in KB \cap C \times P \times R$. Each resource $r \in R$ has a set of lexicalisations, $L_r \subset L$. Lexicalisations are retrieved from the KB, where they are represented as the name or alias, i.e. less frequent name of a resource.

3.1 Seed Selection

Before using the automatically labelled corpus to train a classifier, we detect and discard examples containing highly ambiguous lexicalisations. We measure the degree to which a lexicalisation $l \in L_o$ of an object o is ambiguous by the number of senses the lexicalisation has. We measure the number of senses by the number of unique resources representing a lexicalisation.

Ambiguity Within An Entity Our first approach is to discard lexicalisations of objects if they are ambiguous for the subject entity, i.e. if a subject is related to two different objects which have the same lexicalisation, and express two different relations. To illustrate this, let us consider the problem outlined in the introduction again: *Let It Be* can be both an *album* and a *track* of the subject entity *The Beatles*, therefore we would like to discard *Let It Be* as a seed for the class *Musical Artist*.

Unam: For a given subject s, if we discover a lexicalisation for a related entity, i.e. $(s, p, o) \in KB$ and $l \in L_o$, then, since it may be the case that $l \in L_r$ for some $R \ni r \neq o$, where also $(s, q, r) \in KB$ for some $q \in P$, we say in this case that l has a "sense" o and r, giving rise to ambiguity. We then define A_l^s, the ambiguity of a lexicalisation with respect to the subject as follows: $A_l^s = |\{r \mid l \in L_o \cap L_w \wedge (s, p, o) \in KB \wedge (s, v, w) \in KB \wedge w \neq o\}|$.

Ambiguity Across Classes In addition to being ambiguous for a subject of a specific class, lexicalisations of objects can be ambiguous across classes. Our assumption is that the more senses an object lexicalisation has, the more likely it is that that object occurence is confused with an object lexicalisation of a different property of any class. An example for this are common names of book authors or common genres as in the sentence "*Jack* mentioned that he read *On the Road*", in which *Jack* is falsely recognised as the author Jack Kerouac.

Stop: One type of very ambiguous words with many senses are stop words. Since some objects of relations in our training set might have lexicalisations which are stop words, we discard those lexicalisations if they appear in a stop word list. We use the one described in Lewis et al. [11], which was originally created for the purpose of information retrieval and contains 571 highly frequent words.
Stat: For other highly ambiguous lexicalisations of object entities our approach is to estimate cross-class ambiguity, i.e. to estimate how ambiguous a lexicalisation of an object is compared with other lexicalisations of objects of the same relation. If its ambiguity is comparatively low, we consider it a reliable seed, otherwise we want to discard it. For the set of classes under consideration, we know the set of properties that apply, $D \subset P$ and can retrieve the set $\{o \mid (s, p, o) \in KB \wedge p \in D\}$, and retrieve the set of lexicalisations for each member, L_o. We then compute A_o, the number of senses for every lexicalisation of an object L_o, where $A_o = |\{o \mid \in L_o\}|$.
We view the number of senses of each lexicalisation of an object per relation as a frequency distribution. We then compute min, max, median ($Q2$), the lower ($Q1$) and the upper quartile ($Q3$) of those frequency distributions and compare it to the number of senses of each lexicalisation of an object. If $A_l > Q$, where Q is either $Q1$, $Q2$ or $Q3$ depending on the model, we discard the lexicalisation of the object.

3.2 Relaxed Setting

In addition to increasing the precision of distantly supervised systems by filtering seed data, we also experiment with increasing recall by changing the method for creating test data. Instead of testing, for every sentence, if the sentence contains a lexicalisation of the subject and one additional entity, we relax the former restriction. We make the assumption that the subject of the sentence is mostly consistent within one paragraph as the use of paragraphs usually implies a unit of meaning, i.e. that sentences in one paragraph often have the same subject. In practice this means that we first train classifiers using the original assumption and then, for testing, instead of only extracting information from sentences which contain a lexicalisation of the subject, we also extract information from sentences which are in the same paragraph as a sentence which contains a lexicalisation of the subject. Our new relaxed distant supervision assumption is then:

> If two entities participate in a relation, any *paragraph* that contains those two entities might express that relation, even if not in the same sentence, provided that another sentence in the paragraph in itself contains a relationship for the same subject.

This means, however, that we have to resolve the subject in a different way, e.g. by searching for a pronoun which is coreferent with the subject mention in a different sentence. We use a simpler, less expensive approach: we do not attempt to find the subject of the sentence at all, but instead disregard all features which require the position of the subject mention to be known. Features used in both the relaxed setting and the normal setting are documented in Section 4.6.

Class	Property	Class	Property
Book	author	Film	release date
	characters		director
	publication date		producer
	genre		language
	ISBN		genre
	original language		actor
			character
Musical Artist	album	Politician	birthdate
	active (start)		birthplace
	active (end)		educational institution
	genre		nationality
	record label		party
	origin		religion
	track		spouses

Table 1. Freebase classes and properties used

4 System

4.1 Corpus

To create a corpus for Web relation extraction using background knowledge from Linked Data, four Freebase classes and their six to seven most prominent properties are selected, as shown in Table 1. To avoid noisy training data, we only use entities which have values for all of those properties and retrieve them using the Freebase API. This resulted in 1800 to 2200 entities per class. For each entity, at most 10 Web pages were retrieved via the Google Search API using the search pattern "'*subject_entity*" *class_name relation_name*', e.g. "'The Beatles" Musical Artist Origin'. By adding the class name, we expect the retrieved Web pages to be more relevant to our extraction task. Although subject entities can have multiple lexicalisations, Freebase distinguishes between the most prominant lexicalisation (the entity name) and other lexicalisations (entity aliases). We use the entity name for all of the search patterns. In total, the corpus consists of 560,000 pages drawn from 45,000 different websites. An overview of the distribution of websites per class is given in Table 2.

4.2 NLP Pipeline

Text content is extracted from HTML pages using the Jsoup API [1], which strips text from each element recursively. Each paragraph is then processed with Stanford CoreNLP [2] to split the text into sentences, tokenise it, annotate it with part of speech (POS) tags and normalise time expressions. Named entities are classified using the 7 class (time, location, organisation, person, money, percent, date) named entity model.

[1] http://jsoup.org
[2] http://nlp.stanford.edu/software/corenlp.shtml

Class	%	Website	Class	%	Website
Book	20%	en.wikipedia.org	Film	15%	en.wikipedia.org
	15%	www.goodreads.com		15%	www.imdb.com
	12%	www.amazon.com		3%	www.amazon.com
	9%	www.amazon.co.uk		3%	www.rottentomatoes.com
	4%	www.barnesandnoble.com		1%	www.amazon.co.uk
	3%	www.abebooks.co.uk		1%	www.tcm.com
	2%	www.abebooks.com		1%	www.nytimes.com
	28%	Others		61%	Others
Musical	21%	en.wikipedia.org	Politician	17%	en.wikipedia.org
Artist	6%	itunes.apple.com		4%	www.huffingtonpost.com
	5%	www.allmusic.com		3%	votesmart.org
	4%	www.last.fm		3%	www.washingtonpost.com
	3%	www.amazon.com		2%	www.nndb.com
	2%	www.debate.org		2%	www.evi.com
	2%	www.reverbnation.com		2%	www.answers.com
	57%	Others		67%	en.wikipedia.org

Table 2. Distribution of websites per class in the Web corpus sorted by frequency

4.3 Relation candidate identification

Some of the relations we want to extract values for cannot be categorised according to the 7 classes detected by the Stanford NERC and are therefore not recognised. An example for this is *MusicalArtist:album*, *MusicalArtist:track* or *MusicalArtist:genre*. Therefore, as well as recognising named entities with Stanford NERC as relation candidates, we also implement our own NER, which only recognises entity boundaries, but does not classify them.

To detect entity boundaries, we recognise sequences of nouns and sequences of capitalised words and apply both greedy and non-greedy matching. The reason to do greedy as well as non-greedy matching is because the lexicalisation of an object does not always span a whole noun phrase, e.g. while 'science fiction' is a lexicalisation of an object of *Book:genre*, 'science fiction book' is not. However, for *MusicalArtist:genre*, 'pop music' would be a valid lexicalisation of an object. For greedy matching, we consider whole noun phrases and for non-greedy matching all subsequences starting with the first word of the those phrases, i.e. for 'science fiction book', we would consider 'science fiction book', 'science fiction' and 'book' as candidates. We also recognise short sequences of words in quotes. This is because lexicalisation of objects of *MusicalArtist:track* and *MusicalArtist:album* often appear in quotes, but are not necessarily noun phrases.

4.4 Annotating Sentences

The next step is to identify which sentences express relations. We only use sentences from Web pages which were retrieved using a query which contains the subject of the relation. To annotate sentences, we retrieve all lexicalisations L_s, L_o for subjects and

objects related under properties *P* for the subject's class *C* from Freebase. We then check, for each sentence, if it contains at least two entities recognised using either the Stanford NERC or our own entity recogniser (Section 4.3), one of which having a lexicalisation of a subject and the other a lexicalisation of an object of a relation. If it does, we use this sentence as training data for that property. All sentences which contain a subject lexicalisation and one other entity that is not a lexicalisation of an object of any property of that subject are used as negative training data for the classifier. Mintz et al. [14] only use 1% of their negative training data, but we choose to deviate from this setting because we have less training data overall and have observed that using more negative training data increases precision and recall of the system. For testing we use all sentences that contain at least two entities recognised by either entity recogniser, one of which must be a lexicalisation of the subject. For our relaxed setting (Section 3.2) only the paragraph the sentence is in must contain a lexicalisation of the subject.

4.5 Seed Selection

After training data is retrieved by automatically annotating sentences, we select seeds from it, or rather discard some of the training data, according to the different methods outlined in Section 3.1. Our baseline models do not discard any training seeds.

4.6 Features

Given a relation candidate as described in Section 4.3, our system then extracts the following lexical features and named entity features, some of them also used by Mintz et al. [14]. Features marked with (*) are only used in the normal setting, but not in the relaxed setting(Section 3.2).

- The object occurrence
- The bag of words of the occurrence
- The number of words of the occurrence
- The named entity class of the occurrence assigned by the 7-class Stanford NERC
- A flag indicating if the object or the subject entity came first in the sentence (*)
- The sequence of POS tags of the words between the subject and the occurrence (*)
- The bag of words between the subject and the occurrence (*)
- The pattern of words between the subject entity and the occurrence (all words except for nouns, verbs, adjectives and adverbs are replaced with their POS tag, nouns are replaced with their named entity class if a named entity class is available) (*)
- Any nouns, verbs, adjectives, adverbs or named entities in a 3-word window to the left of the occurrence
- Any nouns, verbs, adjectives, adverbs or named entities in a 3-word window to the right of the occurrence

Compared with the system we use a baseline [14] we use richer feature set, specifically more bag of words features, patterns, a numerical feature and a different, more fine-grained named entity classifier.
We experiment both with predicting properties for relations, as in Mintz et al. [14], and

with predicting properties for relation mentions. Predicting relations means that feature vectors are aggregated for relation tuples, i.e. for tuples with the same subject and object, for training a classifier. In contrast, predicting relation mentions means that feature vectors are not aggregated for relation tuples. While predicting relations is sufficient if the goal is only to retrieve a list of values for a certain property, and not to annotate text with relations, combining feature vectors for distant supervision approaches can introduce additional noise for ambiguous subject and object occurrences.

4.7 Models

Our models differ with respect to how sentences are annotated for training, how positive training data is selected, how negative training data is selected, which features are used, how and if features are combined, and how sentences are selected for testing.
Mintz: This group of models follows the setting of the model which only uses lexical features described in Mintz et al. [14]. Sentences are annotated using the Stanford NERC [8] to recognise subjects and objects of relations, 1% of unrelated entities are used as negative training data and a basic set of lexical features is used. If the same relation tuple is found in several sentences, feature vectors extracted for those tuples are aggregated. For testing, all sentences containing two entities recognised by the Stanford NERC are used.
Comb: This group of models follows the setting described in Section 4. It uses sentences annotated with both Stanford NERC and our named entity recogniser (Section 4.3). All negative training data is used and feature vectors for the same relation tuples are aggregated. For testing, all sentences containing two entities recognised by both Stanford NERC and our named entity recogniser are used.
Sing: The setting for Sing is the same as the setting for Comb apart from that for Sing we do not aggregate feature vectors. This means we predict labels for relation mentions instead of for relations.
Unam, Stop, Stat: Those models select seed data according to the different strategies outlined in Section 3.1.
NoSub: This group of models uses the relaxed setting described in Section 3.2 which does not require sentences to explicitly contain subjects.

4.8 Classifier

In order to be able to compare our results, we choose the same classifier as in Mintz et al. [14], a multi-class logistic regression classifier. We train one classifier per class and model. The models are then used to classify each relation value candidate into one of the relations of the class or NONE (no relation).

5 Evaluation

To evaluate our models we carried out a hold-out evaluation on 50% of our corpus, i.e. for both training and testing we use relations already present in Freebase to annotate our Web corpus. We then conduct an evaluation using those labels for the whole evaluation

set and an additional manual evaluation of the highest ranked 10% of predictions per property [3]. We use the three metrics: number of predictions (number of occurrences which are predicted to be a value of one of the properties for an entity), precision and relative recall. Ideally, we would like to report recall, which is defined as the number of detected true positives devided by the number of positive instances. However, this would mean having to manually examine the whole corpus for every positive instance. Our respective models are restricted as to how many positive predictions they can make by the distant supervision assumption or the relaxed distant supervision assumption. Therefore, we report relative recall for which the number of positive instances equals the number of positive instances identified by automatic labelling.

5.1 Evaluation method

We compute the number of predictions, precision and relative recall for the whole evaluation set and different model combinations using the automatic labels. While this does not allow us to compute exact results for every model, it is a close estimate and is helpful for feature tuning. Results for different models detailed in Section 4.7 averaged over all properties of each class are listed in Table 3. Model settings are incremental, i.e. the row Mintz lists results for the model Mintz, the row after that, + Stop lists results for the model Mintz using the seed selection method Stop, the row after that lists results for the seed selection methods Stop and Unam, and so forth.

For our manual evaluation, we rank all predictions by probability per property and manually annotate and compare from the top 10%, then average results over all properties per class, as shown in Table 4.

5.2 Results

From our automatic evaluation (Table 3) results we can observe that there is a significant difference in terms of performance between the different model groups.
The **Mintz** baseline model we re-implemented has the highest relative recall out of all models. The reason for this is that, for candidate identification, only entities recognised with the Stanford NERC are used. For other models we also use our own named entity recogniser, which does not assign a label to instances. This makes it much more difficult to predict a label because one of the features for the vector (the NE class) is missing. However, the Mintz baseline model also has the lowest precision and the Mintz group of models has the lowest number of positive predictions. The low number of positive predictions is directly related to the low number of relation candidates because the Stanford NERC fails to recognise some of the entities in the text.
The **Comb** group of models has a much higher precision than the Mintz group of models. This difference can be explained by the difference in features, but mostly the fact that the Mintz group of models only uses 1% of available negative training data. The absolute number of correctly recognised property values in the text is about 5 times as high as the Mintz group of features which, again, is due to the fact that Stanford NERC fails

[3] Our evaluation data is available via `www.dcs.shef.ac.uk/~Isabelle/EKAW2014/`

Model	Book			Musical Artist			Film			Politician		
	N	**P**	**R**	**N**	**P**	**R**	**N**	**P**	**R**	**N**	**P**	**R**
Mintz	1248	0.205	**0.844**	2522	0.217	0.716	1599	0.237	**0.722**	1498	0.165	**0.865**
+ Stop	1248	0.204	0.842	2513	0.222	**0.691**	1597	0.236	0.72	1491	0.184	0.764
+ Unam	1258	0.204	0.842	2512	0.230	0.678	1597	0.236	0.72	1490	**0.185**	0.767
+ Stat75	1224	0.234	0.62	2409	0.220	0.277	1582	**0.241**	0.43	1462	0.144	0.514
+ Stat50	1221	**0.240**	0.627	2407	0.232	0.262	1549	0.24	0.4	1459	0.146	0.517
+ Stat25	1205	**0.240**	0.623	2398	**0.250**	0.244	1510	0.22	0.34	1455	0.153	0.515
Comb	1647	0.736	**0.326**	5541	0.619	**0.328**	2506	0.726	**0.403**	1608	**0.809**	**0.513**
+ Stop	1648	0.736	0.311	5516	0.652	0.281	2514	0.723	0.388	1688	0.81	0.476
+ Unam	1648	0.732	0.308	5505	0.65	0.262	2514	0.723	0.388	1674	0.806	0.464
+ Stat75	1622	**0.784**	0.206	5133	0.664	0.136	2505	**0.736**	0.27	1646	0.8	0.3
+ Stat50	1627	0.781	0.204	5130	0.668	0.126	2490	0.735	0.262	1661	0.8	0.29
+ Stat25	1610	0.777	0.182	5053	**0.679**	0.107	2482	0.735	0.241	1662	0.8	0.27
Sing	16242	0.813	**0.476**	19479	0.619	**0.298**	12139	0.726	**0.435**	4970	0.851	**0.653**
+ Stop	16188	0.814	0.46	19213	0.64	0.271	12139	0.726	**0.435**	4952	**0.856**	0.628
+ Unam	16188	0.814	0.46	19162	0.657	0.264	12139	0.726	**0.435**	4952	**0.856**	0.628
+ Stat75	15182	**0.849**	0.288	17204	0.723	0.118	12056	**0.738**	0.321	4896	0.791	0.185
+ Stat50	19072	0.837	0.26	16996	0.729	0.113	12042	0.736	0.302	4897	0.794	0.182
+ Stat25	19239	0.84	0.226	16705	**0.738**	0.101	12003	0.735	0.38	4896	0.795	0.174
Comb NoSub	7523	0.661	0.237	24587	0.595	0.371	10563	0.574	**0.427**	4035	0.633	**0.375**
Sing NoSub	43906	**0.747**	**0.438**	96012	**0.643**	**0.479**	29214	**0.665**	0.359	40848	**0.683**	0.193

Table 3. Automatic evaluation results: Number of positive predictions (N), relative recall (R) and precision (P) for all models and Freebase classes

to recognise some of the relevant entities in the text. However, this also means that the relative recall is lower because those entities are harder to recognise.

We achieve the highest precision overall, though only by a small margin compared to Comb models and dependent on the class, with our **Sing** group of models, which do not combine feature vectors for relation mentions with the same lexicalisation. Because lexicalisations are ambiguous, merging them can lead to noisy feature vectors and a lower precision. On the other hand, rich feature vectors can provide an advantage over sparse feature vectors if they are not noisy.

For the **Unam, Stop and Stat** models, we observe that removing some of the ambiguities helps to improve the precision of models. However, removing too many positive training instances hurts precision. Further, while Stop-Unam improves results for all classes, Stop-Unam-Stat75 does not improve precision for Politician. This model works better for some properties than others due to the original motivation: to improve precision for n-ary properties which on average have multiple values for a property per entity. Although we examine n-ary properties for Politician, all of those have on average just one or two values per property. Therefore, removing positive training examples does not improve precision. For other classes, we achieve the highest precision with the Unam-Stop-Stat75 models, though this comes at the expense of recall, which might not be desirable for some scenarios.

Our **NoSub** models, which are based on a relaxed distant supervision assumption, show a surprisingly high precision. In addition, the total number of positive predictions for

Model	Book		Musical Artist		Film		Politician	
	N	**P**	**N**	**P**	**N**	**P**	**N**	**P**
Mintz	105	0.236	216	0.255	110	0.343	103	0.241
Comb	168	0.739	510	0.672	283	0.764	150	0.863
Sing	1546	0.855	2060	0.586	1574	0.766	488	0.868
Sing Stop-Unam	1539	0.857	2032	0.620	1574	0.766	485	**0.874**
Sing Stop-Unam-Stat75	1360	**0.948**	1148	**0.694**	303	**0.775**	474	0.82
Comb NoSub	705	0.653	2363	0.619	973	0.623	363	0.687
Sing NoSub	4948	0.663	11286	0.547	2887	0.673	3970	0.703

Table 4. Manual evaluation results: Number of true positives (N) and precision (P) for all Freebase classes

the models based on the relaxed assumption is three times as much as for the same models which are based on the original distant supervision assumption. Results based on automatically generated labels for this group of models have to viewed with caution though: automatic labelling is more prone to false positives than for other models.
Our manual evaluation of the highest ranked 10% of results per property (Section 4) confirm the general tendency we already observed for our automatic evaluation. In addition, we can observe that there is a sizable difference in precision for different properties and classes. It is easiest to classify numerical values correctly, followed by people. Overall, we achieve the lowest precision for *Musical Artist* and the highest for *Book*.
When examining the training set we further observe that there seems to be a strong correlation between the number of training instances and the precision for that property. This is also an explanation as to why removing possibly ambiguous training instances only improves precision up to a certain point: the classifier is better at dealing with noisy training data than too little training data.
We also analyse the test data to try to identify patterns of errors. The two biggest groups of errors are entity boundary recognition and subject identification errors. An example for the first group is the following sentence:

"<s>The Hunt for Red October</s> remains a masterpiece of military <o>fiction</o>."

Although "fiction" would be correct result in general, the correct property value for this specific sentence would be "military fiction". Our NER suggests both as possible candidates (since we employ both greedy and non-greedy matching), but the classifier should only classify the complete noun phrase as a value of *Book:genre*. There are several reasons for this: "military fiction" is more specific than "fiction", and since Freebase often contains the general category ("fiction") in addition to more fine-grained categories, we have more property values for abstract categories to use as seeds for training than for more specific categories. Second, our Web corpus also contains more mentions for broader categories than for more specific ones. Third, when annotating training data, we do not restrict positive candidates to whole noun phrases, as explained in Section 4.2. As a result, if none of the lexicalisations of the entity match the whole noun phrase, but there is a lexicalisation which matches part of the phrase, we use that for training and the classifier learns wrong entity boundaries. The second big group of errors is that occurrences are classified for the correct relation, but the wrong subject.

"<s>Anna Karenina</s> is also mentioned in <o>R. L. Stine</o>'s Goosebumps series Don't Go To Sleep."

In that example, "R. L. Stine" is predicted to be a property value for *Book:author* for the entity "Anna Karenina". This happens because, at the moment, we do not take into consideration that two entities can be in *more than one* relation. Therefore, the classifier learns wrong, positive weights for certain contexts.

6 Discussion and Future Work

In this paper, we have documented and evaluated a distantly supervised class-based approach for relation extraction from the Web. Previous distantly supervised approaches have been tailored towards extraction from narrow domains, such as news and Wikipedia, and are therefore not fit for Web relation extraction: they fail to identify named entities correctly, they suffer from data sparsity, and they either do not try to resolve noise caused by ambiguity or do so at a significant increase of runtime. They further assume that every sentence may contain any entity in the knowledge base, which is very costly.
Our research has made made a first step towards achieving those goals. We experiment with a simple named entity recogniser, which we use in addition to a named entity classifier trained for the news domain and find that can especially improve on the number of extractions for non-standard named entity classes such as *MusicalArtist:track* and *MusicalArtist:album*. At the moment, our NER only recognises, but does not classify named entities. In future work, we aim to research distantly supervised named entity classification methods to assist relation extraction.
To overcome data sparsity and increase the number of extractions, we experiment with relaxing the distant supervision assumption to extract relations across sentence boundaries. We find that this results in about six times the number of extractions at a still fairly reasonable precision. Our future plans to improve on those results are to research unsupervised Web-based coreference resolution methods. One additional resource we want to exploit for this is semi-structured information contained on Web pages, since it is much easier to interpret than free text.

References

1. Alfonseca, E., Filippova, K., Delort, J.Y., Garrido, G.: Pattern Learning for Relation Extraction with a Hierarchical Topic Model. In: Proceedings of ACL. Jeju, South Korea (2012)
2. Augenstein, I.: Seed Selection for Self-Supervised Web-Based Relation Extraction. Proceedings of the 3rd Workshop on Semantic Web and Information Extraction (2014), to appear
3. Augenstein, I., Padó, S., Rudolph, S.: LODifier: Generating Linked Data from Unstructured Text. In: Proceedings of ESWC. pp. 210–224 (2012)
4. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: A Collaboratively Created Graph Database For Structuring Human Knowledge. In: Proceedings of ACM SIGMOD. pp. 1247–1250 (2008)
5. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E.R., Mitchell, T.M.: Toward an Architecture for Never-Ending Language Learning. In: In AAAI (2010)
6. Del Corro, L., Gemulla, R.: ClausIE: Clause-Based Open Information Extraction. In: Proceedings of the 22nd International Conference on World Wide Web. pp. 355–366 (2013)

7. Fader, A., Soderland, S., Etzioni, O.: Identifying relations for open information extraction. In: Proceedings of EMNLP. pp. 1535–1545. Association for Computational Linguistics (2011)
8. Finkel, J.R., Grenager, T., Manning, C.D.: Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In: Proceedings of ACL (2005)
9. Gerber, D., Ngomo, A.C.N., Gerber, D., Ngomo, A.C.N., Unger, C., Bühmann, L., Lehmann, J., Ngomo, A.C.N., Gerber, D., Cimiano, P.: Extracting Multilingual Natural-Language Patterns for RDF Predicates. In: Proceedings of EKAW. pp. 87–96 (2012)
10. Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L.S., Weld, D.S.: Knowledge-Based Weak Supervision for Information Extraction of Overlapping Relations. In: Proceedings of ACL. pp. 541–550 (2011)
11. Lewis, D.D., Yang, Y., Rose, T.G., Li, F.: Rcv1: A new benchmark collection for text categorization research. Journal of Machine Learning Research 5(Apr), 361–397 (2004)
12. Mausam, Schmitz, M., Soderland, S., Bart, R., Etzioni, O.: Open Language Learning for Information Extraction. In: Proceedings of EMNLP-CoNLL. pp. 523–534 (2012)
13. Min, B., Grishman, R., Wan, L., 0001, C.W., Gondek, D.: Distant Supervision for Relation Extraction with an Incomplete Knowledge Base. In: Proceedings of HLT-NAACL. pp. 777–782 (2013)
14. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proceedings of ACL. vol. 2, pp. 1003–1011 (2009)
15. Nakashole, U., Theobald, M., Weikum, G.: Scalable Knowledge Harvesting with High Precision and High Recall. In: Proceedings of WSDM. pp. 227–236 (2011)
16. Nguyen, T.V.T., Moschitti, A.: End-to-End Relation Extraction Using Distant Supervision from External Semantic Repositories. In: Proceedings of ACL (Short Papers). pp. 277–282 (2011)
17. Presutti, V., Draicchio, F., Gangemi, A.: Knowledge Extraction Based on Discourse Representation Theory and Linguistic Frames. In: Proceedings of EKAW. pp. 114–129 (2012)
18. Riedel, S., Yao, L., McCallum, A.: Modeling Relations and Their Mentions without Labeled Text. In: Proceedings of ECML PKDD. pp. 148–163 (2010)
19. Roth, B., Klakow, D.: Combining Generative and Discriminative Model Scores for Distant Supervision. In: Proceedings of ACL-EMNLP. pp. 24–29 (2013)
20. Suchanek, F.M., Kasneci, G., Weikum, G.: YAGO: A Large Ontology from Wikipedia and WordNet. Web Semantics: Science, Services and Agents on the World Wide Web 6(3), 203–217 (2008)
21. Surdeanu, M., Tibshirani, J., Nallapati, R., Manning, C.D.: Multi-instance Multi-label Learning for Relation Extraction. In: Proceedings of EMNLP-CoNLL. pp. 455–465 (2012)
22. Takamatsu, S., Sato, I., Nakagawa, H.: Reducing Wrong Labels in Distant Supervision for Relation Extraction. In: Proceedings of ACL. pp. 721–729 (2012)
23. Unger, C., Bühmann, L., Lehmann, J., Ngonga Ngomo, A.C., Gerber, D., Cimiano, P.: Template-Based Question Answering over RDF Data. In: Proceedings of WWW. pp. 639–648 (2012)
24. Vrandečić, D., Krötzsch, M.: Wikidata: A Free Collaborative Knowledge Base. Commun. ACM (2014), to appear
25. Xu, W., Hoffmann, R., Zhao, L., Grishman, R.: Filling Knowledge Base Gaps for Distant Supervision of Relation Extraction. In: Proceedings of ACL. pp. 665–670 (2013)
26. Yao, L., Riedel, S., McCallum, A.: Collective Cross-document Relation Extraction Without Labelled Data. In: Proceedings of EMNLP. pp. 1013–1023 (2010)
27. Yates, A., Cafarella, M., Banko, M., Etzioni, O., Broadhead, M., Soderland, S.: TextRunner: Open Information Extraction on the Web. In: Proceedings of HLT-NAACL: Demonstrations. pp. 25–26 (2007)