EUSKAL HERRIKO UNIBERTSITATEA
University of the Basque Country

PhD thesis summary

Knowledge Base Population through Distant Supervision: Analysis and Improvements

Ander Intxaurrondo Gonzalez de Langarika

2015

EUSKAL HERRIKO UNIBERTSITATEA
University of the Basque Country

Knowledge Base Population through Distant Supervision: Analysis and Improvements

This summary is a shortened and translated version of the dissertation entitled "Ezagutza-baseen aberasketa urruneko gainbegiraketaren bidez: analisiak eta hobekuntzak", written by Ander Intxaurrondo under the supervision of Dr. Eneko Agirre and Dr. Oier Lopez de Lacalle. It also includes the papers which the candidate has published on the research presented here.

March 2015

Acknowledgments

The Department of Education, Universities and Research of the Basque Government who awarded a pre-doctoral fellowship (BFI06.281) to the author of this PhD dissertation to conduct this research.

Abstract

Information extraction consists on getting structured information automatically from texts. Information extraction systems try to find relevant information at corpora, and return a representation of the information in an intuitive way for both humans and computers. In this dissertation we focus on two of its sub-tasks: relation extraction, which consist on the identification of relations between entities and their attributes, and event extraction, which consists on identifying events in free text and deriving detailed and structured information about them.

In distant supervision, if a pair of entities participates in relation in a knowledge base, all sentences containing that entity-pair expresses that relation somehow. Relation extraction methods based on distant supervision rely on true tuples to retrieve noisy mentions, which are then used to train traditional supervised relation extraction methods. In this dissertation, we have analyzed the sources of noise in the mentions, and explore methods to filter out noisy mentions. The results show that a combination of our heuristics is able to significantly outperform two strong baselines.

In addition, we introduce a distantly supervised event extraction approach that extracts complex event templates from microblogs. This near real-time data source contains information that is both approximate and ambiguous, impacting both the evaluation and extraction methods. About the former, we devise a lenient evaluation measures that incorporates similarity between extracted values and the gold truth, giving partial credit to different but similar values. With respect to extraction, we directly address approximate information, including positive training examples that contain information similar but not identical to gold values. We positively evaluate our contributions on the complex domain of earthquakes, with events with up to 20 arguments. The dataset containing the knowledge base, relevant tweets and manual annotations is publicly available.

Contents

1 Introduction

1.1 Information extraction

Information extraction, also known as *text analytics* commercially, consists on getting structured information automatically from texts. Information extraction systems try to find relevant information at corpora, and return a representation of the information in an intuitive way for both humans and computers.

With information extraction, we can learn drug-gene product interactions from medical research literature; places of birth, jobs, etc. from people biographies; earnings, profits, board members, headquarters, etc. from company reports; or casualties, damages and wind velocities from tornado reports.

Information extraction is usually divided in several sub-tasks, as follows:

- **Named entity recognition (NER):** systems find and classify names in texts, such as people names, organizations, locations, temporal expressions or numerical expressions:
  - The decision by the independent MP **Andrew Wilkie** [*person*] to withdraw his support for the minority **Labor** [*organization*] government sounded dramatic but it should not further threaten its stability.[1]
- **Co-reference resolution:** detection of co-reference and anaphoric links between text entities:
  - **David Beckham** won't be appearing in his fourth World Cup, though. **The 35-year-old midfielder** tore his left Achilles' tendon while playing for AC Milan on March 14 and will miss the entire tournament.[2]
- **Terminology extraction:** automatically extract relevant terms from a given corpus:
  - **Latent semantic analysis** is a technique in natural language processing, in particular in **vectorial semantics**, of **analyzing relationships** between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms.[3]
- **Relation extraction:** identification of relations between entities and their attributes:

[1] Text taken from: http://www.smh.com.au/federal-politics/political-opinion/wilkie-blow-not-lethal-for-labor-20120122-1qc9d.html

[2] Text taken from http://sports.espn.go.com/espn/wire?section=soccer&id=5278599

[3] http://labs.translated.net/terminology-extraction/?esempio=6

  - **Clint Eastwood** was born in **San Francisco** and is divorced to **Dina Eastwood**.
    Extracted relations:
    * Clint Eastwood - *born_in* - San Francisco
      (PERSON - *born_in* - LOCATION)
    * Clint Eastwood - *divorced_to* - Dina Eastwood
      (PERSON - *divorced_to* - PERSON)

- **Event Extraction:** identifying events in free text and deriving detailed and structured information about them:
  - **200,000** people start protesting in **Pakistan**.[4]
    Extracted event information:
    * *Type:* Conflict-Demonstrate
    * *Anchor:* protesting
    * *Place:* Pakistan
    * *Entity:* 200,000

Information extraction is very useful for many other NLP applications, such as text simplification (Klebanov et al. 2004), augmenting current knowledge bases such as Freebase or DBpedia (Mintz et al. 2009), question-answering (Ravichandran and Hovy 2002), text summarization (Mihalcea and Tarau 2004), machine translation (Babych 2005), opinion mining (Pronoza et al. 2014), protein iterations (Miyanishi and Ohkawa 2013), etc.

This dissertation focuses on **relation extraction** and **event extraction** for **knowledge base population**. These tasks are explained below.

Relation extraction

Relation extraction detects and classifies semantic relations between entities or attributes in text, such as *located-in, employed-by, part-of* or *married-to* and many more. For instance, if a relation extraction system receives the following sentence:

- The headquarters of **BHP Billiton Limited** are located in **Melbourne, Australia**.[5]

The system extracts that BHP Billiton Limited's headquarters are in the Australian city of Melbourne, following a structured representation, such as:

[4] Example taken from the ACE 2005 corpus, document file *CNN_CF_20030303.1900.06-2.apf.xml*

[5] https://web.stanford.edu/class/cs124/lec/Information_Extraction_and_Named_Entity_Recognition.pdf

- BHP Billiton Limited - *headquarters_in* - Melbourne, Australia
  (COMPANY - *headquarters_in* - LOCATION)

Above, we extracted a relation triplet. Relation triplets are formed by two entities and the relation between them. The format used above is *"subject - relation - object"*, we will use this format every time we refer about an specific relation.

The *subject* is the **main entity** of the relation. The *object* is not always an entity, as it can refer to attributes. Attributes can be occupations, religions or titles for people, and websites for organizations. To simplify explanations during this dissertation, we will refer about entities in general, and mention attributes where necessary. Most of the relation extraction systems focus on extracting binary relations, but it is possible to find higher-order relations as well, although they are not very common (Bach and Badaskar 2007). There are different ways to build relation extractors: using hand-written patterns; or using supervised, unsupervised or semi-supervised machine learning.

This dissertation focuses on **distant supervision**, a semi-supervised technique, which is introduced in section 1.2. The relation types used at our experiments are all binary.

Event extraction

Event extraction aims to extract information about the role of different elements found at the same context play in an event. Events can be related, among others, to business (companies merged, bankruptcy,...), conflicts (demonstrations, countries attacked,...), justice (arrests, trail sentences,...), or natural disasters (earthquakes, hurricanes,...). If we analyze the following sentence about airplane crashes:

- **20** dead, **15** injured in a **US Airways Boeing 747** crash.

We can extract the following information about the event:

- *Event Type:* Airplane Crash
- *Fatalities:* 20
- *Injured:* 15
- *Operator:* US Airways
- *Aircraft Type:* Boeing 747

In this dissertation, we extract information about events related to earthquakes using distant supervision.

Knowledge Base Population

A knowledge base stores complex structured information in a computer system. We will focus on large knowledge bases, such as Freebase[6] and DBpedia[7], that are created semi-automatically from the infoboxes in Wikipedia articles.

Knowledge Base Population is the task of taking an incomplete knowledge base and a large corpus of text, and completing the missing elements of the knowledge base. That is, the computer has to "read" the text and get information out of it. The task can be done building a knowledge base from scratch, or populating and updating an existing knowledge base with missing information. We will focus on the second: assume that the structure of the knowledge base is given, and that the relations in the knowledge base are partially populated, with missing values.

The Knowledge Base Population (KBP) shared task, is conducted by NIST as part of the Text Analysis Conference. This task aims to bridge the information extraction and question answering communities to promote research in discovering facts about entities, and expanding a knowledge base with new facts. KBP is done through two separated sub-tasks (Ji and Grishman 2011):

- **Entity Linking:** Extract name-entities from text and link them with its corresponding entry in the knowledge base.

- **Slot Filling:** Collect information from the corpus regarding certain relations of an entity and populate the knowledge base.

During the dissertation, we participated at the 2011 **Slot Filling** task. This task is also used as a reference for different experiments at this dissertation.

1.2 Distant supervision

Distant supervision is a paradigm proposed by Mintz et al. (2009) for relation extraction systems. The approach consists on aligning existing information in a knowledge base with unlabeled text. The algorithm labels the information at corpora automatically, and combines the advantages of both supervised and unsupervised learning algorithms used for relation extraction. One of the main objectives of distant supervision is to avoid the manual annotation of corpora. This approach is domain independent, and it has been mostly used to extract information about people, organizations and locations. According to Mintz et al.:

[6] http://www.freebase.com/
[7] http://dbpedia.org/About

Figure 1 – Aretha Franklin's information at Freebase.

> "The intuition of distant supervision is that any sentence that contains a pair of entities that participate in a known relation in a knowledge base is likely to express that relation in some way."

The distant supervision algorithm is supervised by a database, and does not suffer from overfitting or domain-dependence that plague supervised systems. Unlike unsupervised approaches, the output of the different classifiers uses canonical names for relations (Mintz et al. 2009).

Nevertheless, we compulsorily need a knowledge base to work with distant supervision. For instance, figure 1 shows part of the information about Aretha Franklin at Freebase[8]. At Freebase, we find the name of the article as the main entity (Aretha Franklin), and many relations extracted from the knowledge base, such as[9]:

- Aretha Franklin - *date_of_birth* - March 25, 1942
- Aretha Franklin - *occupation* - Singer
- Aretha Franklin - *city_of_birth* - Memphis

We also need a corpus to extract the sentences with occurrences of the main entity (Aretha Franklin). Once we get all contexts, we must detect the remaining entities in the context that are stored in the knowledge base. If

[8] http://www.freebase.com/m/012vd6

[9] In order to improve readability, we will use intuitive tags instead of the actual Freebase relation names, i.e. *date_of_birth* for */people/person/date_of_birth*, *city_of_birth* for */people/person/place_of_birth*, and *occupation* for */people/person/profession*.

Sentence	Relation
Aretha Franklin was born on **March 25, 1942**, in Memphis, Tennessee.	*date_of_birth*
Aretha Franklin (born **March 25, 1942**) began as a gospel singer.	*date_of_birth*
Aretha Franklin (born March 25, 1942) began as a gospel **singer**.	*occupation*
(...) failed to show legendary **singer Aretha Franklin** any respect.	*occupation*
Aretha Franklin is a legendary American soul **singer**.	*occupation*
Aretha Franklin was born on March 25, 1942, in **Memphis**, Tennessee.	*city_of_birth*
The 'Queen of Soul' **Aretha Franklin** was born right here in **Memphis**.	*city_of_birth*

Table 1 – Different sentences about Aretha Franklin obtaining pairs of entities with a known relation. The right column indicates the relation between the person and the entity in bold.

one of the detected entities is related to the main entity according to the knowledge base, then we will annotate this sentence with the corresponding relation label. For example, table 1 shows different sentences about the famous singer, which indicate a relation of the examples given above. Note that the contexts may show more than one relation; for example, the second and third sentences in the table show the *date_of_birth* and the *occupation* relation types. We have a similar situation with the first and sixth sentences.

On the other hand, the entities in the context that have no explicit relationship in the knowledge base are considered *unrelated*. For example, there is no relation specified in the knowledge bases between the city of San Francisco and the singer, so the pair in the following sentence will be labeled as *unrelated*:

- **Aretha Franklin** will perform in **San Francisco** next week.

Once we align the knowledge base and the text as above, we perform supervised learning. For training, we create the feature-set for each sentence, which are later provided to a classifier. The learning is done using a supervised relation extraction system, where contexts and models are learned, one per relation type.

We are now ready to apply the models and extract new values from text for entities not included at the training set. We search for entities that are missing in the knowledge base, or have incomplete information, we consider these entities as **target entities**, the entities we are interested to get information for. The following steps explain the testing process:

1. We first search for sentences where the target entity is mentioned and extract them.

2. We detect and mark other entities in the text, which could be potentially related to the target entity.

3. We make entity pairs between the target entity and other entities found in the same sentence.

4. We extract features for each entity pair, and decide if is there any relation between them.

Let's suppose our testing set has the following sentence[10], where Morgan Freeman is our target entity:

- It was on this date, June 1, 1937, American actor, film director, and narrator **Morgan Freeman** was born.

The system has to guess if the underlined information is related or not to the target entity. After analyzing the features of each potential entity, the relation extraction system, when applied to the sentence, will hopefully extract the following relations:

- Morgan Freeman - *date_of_birth* - June 1, 1937
- Morgan Freeman - *occupation* - actor
- Morgan Freeman - *occupation* - director
- Morgan Freeman - *occupation* - narrator

If we take a look at the information extracted from the sentence above about Morgan Freeman, and check if that information is included in knowledge bases such as DBpedia[11] or Freebase[12], we can see that we extracted a new relation that is missing in both knowledge bases:

- Morgan Freeman - *occupation* - narrator

These knowledge bases do not mention that Freeman also worked as narrator in several movies and documentaries. So once we find this new relation, that information can be added to the knowledge base.

1.3 Main difficulties of distant supervision

Distant supervision tries to take the best of supervised and unsupervised approaches. Similarly to supervised approaches, we can represent contexts with rich and complex features. We also have the advantage to work with canonical relation names, because relation names in knowledge bases are normalized, this simplifies the experimental and evaluation phase. On the

[10] Sentence found at website http://freethoughtalmanac.com/?p=6861
[11] http://dbpedia.org:8890/page/Morgan_Freeman
[12] http://www.freebase.com/m/055c8

other hand, similar to unsupervised approaches, we can make use of large amounts of unlabeled data.

Although distant supervision has been well received by the NLP community as an alternative to supervised and unsupervised learning, the approach has several issues which call for improvement. The following list introduces some of the issues we have discovered while working on the dissertation. Most of our contributions consist on solutions to alleviate some of them.

Necessity of negative mentions: Classifiers need to distinguish between mentions that, according to the distant supervision hypothesis, are labeled with a relation (positive mentions), and unrelated (or negative) mentions. We call negative mentions to those where the participating entities are not related, according to the relations in the knowledge base. Negative mentions help the classifier considering when there is no relation between two entities in a given context. They are very necessary if we want distant supervision systems to perform well. All state-of-the-art distant supervision systems integrate datasets with negative mentions.

Lack of positive mentions: Often the frequency of some relations between entity pairs is too low, because some specific relation types are not very common in knowledge bases. Having few mentions about those relations to train, compared to others, it makes difficult to predict new information about the target entities. This problem gets even worse when we do not have mentions for a specific relation type at all. As expected, this makes the system unable to make predictions for that relation. This issue happens in supervised information extraction systems as well. This dissertation does not focus on this issue.

Noisy mentions: Distant supervision gathers a number of noisy mentions. Probably, the noise generated by distant supervision is the most important issue. According to distant supervision, all sentences containing two entities with a relation in the knowledge base will explicitly express that relation, which is not true in all cases. We gave many examples about Aretha Franklin in section 1.2, where that supposition happens, but this is not always true. Thus, accepting these examples we are introducing noise into the system. In this dissertation we recognized three different types of noisy mentions as follows:

1. **Wrong context:** This is the most common noise type in positive mentions, and happens when two entities that are related in the knowledge base, but the context of that mention does not express that relation. For example, table 2 contains two examples of the same entity pairs and relation types showed in table 1, but in this case the context does

Sentence	Relation
Celine Dion is a great **singer** and a good friend of **Aretha Franklin**.	*occupation*
Aretha Franklin gave a great concert in **Memphis** last night.	*city_of_birth*

Table 2 – Wrong context on Aretha Franklin. The right column indicates the intended relation between the entity pain (in bold).

Sentence	Relation
Celso Amorim, the Brazilian **Foreign Minister**, said the (...)	*unrelated*
Celso Amorim served as **Minister of External Relations** of Brazil.	*unrelated*

Table 3 – Negative mentions that should be positives because of an incomplete knowledge base.

not express any relation between the entities. The first sentence talks about Celine Dion, who is a singer like Aretha Franklin, but the relation here is the occupation of Celine Dion. The second sentence is just about a concert Franklin gave at Memphis, where happens to be her birthplace.

2. **Incomplete knowledge base:** These noisy negative mentions are generated when the context of a mention actually expresses a relation between two entities, but the knowledge base does not support that relation. Therefore the mention is labeled as *unrelated* at the dataset. The relation extraction module will incorrectly learn that such patterns do not express any relation, degrading the results. For example, Freebase has incomplete information about Celso Amorim, ex-minister of defense and external relations of Brazil. The knowledge base does not specify his duty at the Brazilian government. Thus, all mentions where Celso Amorim and Brazil participate together will be labeled as *unrelated*, instead of *occupation*. Table 3 shows two examples of negative mentions where the relation *occupation* is explicit.

3. **Multi-label relations:** This phenomena occurs when the knowledge base supports more than one relation for the same entity pair. Given the context that the pair occurs, distant supervision is not able to disambiguate the relation, and labels the context with all the relation types that the knowledge base supports for such case. For instance, let's consider tuple *Rupert Murdock* and *News Corporation*, with relations *founder* and *top_member* between them. Table 4 shows two sentences where both entities appear, and distant supervision tags both examples with both relations (second column) when only one relation is correct (underlined relation label).

Sentence	Relations
News Corporation was founded by **Rupert Murdock**.	**founder** / *top_member*
Rupert Murdock is the CEO of **News Corporation**	*founder* / **top_member**

Table 4 – Examples of multi-label relations, the second column indicates the relations between bolded entities in the knowledge base, bolded relational labels are the correct relations of the sentence.

Noisy mentions will make relation extraction modules to learn from incorrect relation labels, causing incorrect predictions.

In this dissertation, we focus on solving the **wrong context** and **multi-label relations** problems. We assume that the knowledge base is complete in the experiment settings.

Approximate values: There are cases where we encounter information in the knowledge base and the corpus at hand that are both similar, but not identical. While creating the training mentions, when we look for entities and relations between them (if any) at retrieved sentences, we look for mentions that match exactly with the information in the knowledge base. Information at the corpus that is very similar to the one in the knowledge base is excluded.

Inaccurate values, those that are not completely equal with the gold information, still might be useful to learn useful patterns of specific relation types and return better predictions. On the contrary, if the classifier receives two sentences with the same pattern, but one sentence has a relation label, and the other is considered unrelated (because the information did not exactly match with the knowledge base), the classifier will get confused. The differences cause that many related mentions are missed by distant supervision, hurting the performance of the distant supervision system and causing important information loss. This problem does not only happen with distant supervision, we can find this issue in all information extraction tasks.

Suppose a company at the training set, with the knowledge base indicating that the company has 140 employees. An article in the corpus could mention there are 144 employees, probably because the article is more recent than the publication date of the knowledge base. The value 144 would be missed by distant supervision, leaving those sentences as *unrelated*.

Moreover, traditional evaluation procedures penalize predicted values which are very similar to the ones at the gold standard, without taking account how approximate the predicted value is. The information of the knowledge bases is assumed to be perfect, but we cannot guarantee that is always true. Considering the example above, if that company was a target entity, and we wanted to extract relations about it, all predictions where the number of employees was 144 would be considered completely incorrect. An outdated value would be considered valid, and the updated one invalid.

At this dissertation, we propose to use mentions with similar content to the information in the knowledge base. We also propose a soft evaluation metric, which considers similar information as partially correct at the evaluation process.

1.4 Prior work in the IXA NLP Group

This dissertation has been carried out within the IXA NLP group. This research group of the University of the Basque Country has been working on NLP for more than twenty-five years. Even though this group mainly focuses on applied research in the Basque language, it also works on research and development of tools in other languages.

Fernandez (2012) worked on named entities of the Basque language. Her dissertation focused on identifying, classifying, translating and disambiguating named entities. She used supervised, unsupervised and semi-supervised methods at the mentioned tasks, being able to compare the effectiveness of each learning methods for each task. She also analyzed the impact of morphosyntactic features of Basque, when trying to automate the treatment of name-entities.

Urizar (2012) worked with the identification of multi-word lexical units in texts for Basque, designing and developing a system called *HABIL* that helped to analyze them. Meanwhile, Gurrutxaga (2014) worked with the automatic extraction of phraseological units with *noun+verb* format.

The Kyoto Project[13] worked with some knowledge yielding robots, known as *Kybots*. Kybots enriched text with linguistic and semantic information, defined patterns in texts and extracted patterns for all languages. Patterns were created manually. Kybots were initially used to extract information about climate at the Kyoto Project, and they are also used nowadays at the NewsReader Project[14] in other domains.

[13] http://kyoto-project.eu/xmlgroup.iit.cnr.it/kyoto/index.html
[14] http://www.newsreader-project.eu/

2 Outline of the dissertation

In the following, the structure and contents of the original thesis in Basque is briefly explained.

- **Chapter 2** - *State-of-the-art.*

  This chapter is devoted to the description of different knowledge bases, and different learning approaches used for relation extraction. There is a section dedicated to different datasets and shared tasks. We also focus on different research lines about event extraction and distant supervision.

- **Chapter 3** - *Development of a distant supervision system.*

  In this chapter, we make the first steps to develop a distant supervision system for relation extraction. We initially develop a supervised learning system, using a manually annotated corpus from ACE 2005.

  Supervised systems and distant supervision systems are very similar to each other. The most notable difference is the way corpora are labeled, manually in supervised systems, and automatically in distant supervision systems. Mention preproccesing, feature extraction, classification and inference are compatible in both systems.

  Regarding the distant supervision system, we have participated in a shared task based on knowledge base population called "Slot Filling" at TAC 2011. In this system, we extract mentions from documents containing entities related to each other, being that relation specified in a knowledge base. Once we extract all mentions, we apply to our distant supervision system the same features, optimization techniques and inference methods used at the supervised system to get the final results.

  The obtained results are far from satisfactory. Our error analysis shows that one of the reasons why our system gives bad results is the absence of negative examples. Another reason is the lack of positive examples for some relations, making the learning process for those relation types very hard.

  The most important error, is the huge amount of noisy mentions detected in our dataset. We find lots of mentions at our training set, where according to the context, there is no relation between the participating entities, but the mention is labeled with the relation type of the knowledge base. Noisy mentions make the learning process worse, confusing the classifier, returning incorrect answers, and thus questioning the performance of the system. To improve the performance, we need to remove noisy mentions. We also decide to use an existing distant

supervision system, the one developed by the Stanford University NLP group (Surdeanu et al. 2012), which improves the original algorithm, to test our noise filtration methods.

- **Chapter 4** - *Removing noisy mentions for distant supervision.*

  In this chapter, we first analyze a random sample of the dataset created by Riedel et al. (2010), and categorize the noise in three different types.

  Motivated by the huge amount of noisy mentions that are extracted, we present three simple and robust heuristics for noise filtering. These heuristics do not use any manual annotation at the datasets, and are system and domain independent. We combine them in different ways, and final experiments are done using the best combination model.

  We use a state-of-the-art distant supervision system developed by Stanford University (Surdeanu et al. 2012). This system includes two variants that improve the original algorithm presented by Mintz et al.. Using this system helps us evaluate the performance of our heuristics.

  These heuristics, specially their combinations, outperform two strong baselines developed by Stanford University, demonstrating how important is detecting and removing noisy mentions for the performance of distant supervision.

- **Chapter 5** - *Creating a knowledge base and a tweet dataset about earthquakes.*

  In this chapter, we begin with the first steps to create an event extraction system based on tweets. The chosen domain is earthquakes. The first step consists on creating a knowledge base based on the latest earthquakes reported in Wikipedia. Later, we extract tweets related to the earthquakes of the knowledge base, and remove tweets about aftershocks.

  The knowledge base contains information about 108 earthquakes, 20 different argument types, and 1,116 argument values. The dataset contains a collection of relevant tweets about these earthquakes, with 7,841 tweets in total. The mentions in the tweets are also annotated manually, in order to use it further for a deep analysis of the dataset.

  The knowledge base and tweet dataset, alongside manual annotations, are publicly available for free.

- **Chapter 6** - *Exploring distant supervision for event extraction from Twitter.*

  In this chapter, we develop a distant supervision system for event extraction, based on Conditional Random Fields. Our system extracts complex events that work with 20 arguments of different types.

We first analyze the automatically annotated dataset by the distant supervision algorithm, and compare it with the manually annotated version. We estimate that about 83% of the information annotated in the tweets by distant supervision is correct, but only 47% of the information manually annotated matches with the gold information.

We find as well, that the 55% of the predicted answers, when do not match the value in the knowledge base, are very similar to the gold ones. We also see that 20% of the predicted answers have no information in the knowledge base for some arguments, this shows that tweets include interesting information missing from the knowledge base. We propose a lenient evaluation, where similar answers are considered partially correct. Using the lenient evaluation improves the results, as expected, and makes the system evaluation more realistic, specially considering that the information in the knowledge bases is sometimes inexact.

We also propose a new variation for distant supervision, where the mentions at the tweets that are very similar to the gold information are also annotated. We call this "approximate matching". In addition, we apply a global feature aggregation approach to provide richer information between tweets that belong to the same earthquake. Our results on tweets related to earthquakes show that each improvement yields a 19% of significant improvement. We show that these contributions are complementary: a model that combines approximate matching and feature aggregation performs better than each of them individually, with an improvement of 33% over the baseline system. Our results reach 88% of the ceiling performance (the result we get if we train the system using the manually annotated dataset) showing that distant supervision is useful for complex event extraction.

- **Chapter 7** - *Conclusions and future work.*

3 Thesis contributions and future work

In this section, we present the contributions of this work, organized by the chapters they belong to. We finalize presenting the future work.

3.1 Contributions

We now describe the main contributions of this work, including their relation to chapters in the full dissertation:

- **We study the main difficulties of distant supervision:** (*All chapters*)

  We analyze the main difficulties through different experiments. These difficulties are all described at the introductory chapter, subsection 1.3 of this thesis summary, which include:

  1. Problems with the learning process because of the lack of positive mentions. (*Chapter 3*)
  2. The necessity of negative mentions for the learning process. (*Chapter 3*)
  3. Confusion of the system because of noisy mentions. (*Chapter 4*)
  4. Small discrepancies between the information in the knowledge base and the corpus at hand, due to approximate values. (*Chapter 6*)

- **We provide an analysis of different types of noise for distant supervision:** (*Chapter 4*)

  We analyze random mentions at the dataset developed by Riedel et al. (2010). We categorize three different noise types:

  1. **Wrong context:** in positive mentions happens when two entities that are related appear at the same mention, but the context of that mention does not express that relation. This is the most common noise type.
  2. **Incomplete knowledge base:** generated when the context of a mention expresses a relation between two entities, but due to the knowledge base does not support that relation, the mention is labeled as unrelated at the dataset.
  3. **Multi-label relations:** when the knowledge base supports more than one relation for the same entity pair, but only one of them (or even none) is valid according to the context of the sentence they are participating.

- **We propose heuristics to remove wrong contexts in positive mentions:** (*Chapter 4*)

  We present three different heuristics that detect and delete noisy mentions, in order to improve the performance of distant supervision systems:

  1. **Triplets with too many mentions:** triplets with too many mentions tend to have many noisy mentions, we remove those triplets above a certain threshold.
  2. **Pointwise Mutual Information (PMI) between tuples and relations:** those tuples with low PMI tend to have many noisy mentions, we remove those triplets below a certain threshold.
  3. **Similarity between triplets and their relation centroids:** mentions far away from the centroids of the relation label tend to be noisy, we remove mentions further from a threshold.

  A combination of these three heuristics significantly outperforms two strong baselines developed by the Stanford University.

- **We propose a lenient evaluation score for distant supervision:** (*Chapter 6*)

  We propose a lenient evaluation model, where predicted values that are very similar to the gold information, are considered partially correct. This evaluation reflects the performance of relation extraction systems better.

- **We propose an extension of distant supervision to label approximate values:** (*Chapter 6*)

  Distant supervision labels mentions as related if and only if the information found at the sentence is exactly the same as the one in the knowledge base. Mentions that are similar are considered as unrelated, but those mentions could contain good patterns for the learning process of the system.

  We propose considering those mentions with approximate values to those in the knowledge base as positive mentions, to increase the quality of the training dataset. This improves the performance of the system significantly.

- **We build a knowledge base and a dataset for event extraction:** (*Chapter 5*)

  In order to try distant supervision for event extraction, we have created a knowledge base about earthquakes with information extracted from Wikipedia. The knowledge base includes up to 20 different arguments.

We have created a dataset with tweets extracted from Twitter as well, tweets related to the earthquakes of our knowledge base. This knowledge base and dataset have been used for the experiments of chapter 6.

We have also annotated manually all the relevant information we can find at the tweets, where the context matches with the arguments in the knowledge base. The manually annotated dataset has been useful for further analysis. To our knowledge, this is the only distant supervision dataset where a manual annotated version of the mentions is available.

The knowledge base and the dataset, including the manually annotated version of the dataset, are available to the public for free.

- **We apply distant supervision to complex events:** (*Chapters 5 and 6*)

  We experiment with the dataset and the knowledge base explained above. We show that distant supervision is suitable with event extraction. Until now, most distant supervision systems that tried to test the approach with event extraction only worked with 1 or 2 arguments, while our system works up to 20 different arguments.

- **We apply distant supervision to social media:** (*Chapters 5 and 6*)

  At the experiments related to event extraction and distant supervision, the used dataset is built with tweets extracted from Twitter, instead of traditionally used text fragments extracted from newswire documents. Newswire documents contain formal and complex text, and non-ambiguous language. Meanwhile, social media sources are written in informal language, ambiguous info and noisy text fragments.

  Our experiments show that distant supervision performs well with microblogs to fill information of knowledge bases.

3.2 Future Work

There are some open research lines in this work that can be explored further. In this subsection we describe the main experiments and paths to be explored we would like perform in the future.

- **Remove noisy mentions on other datasets:**

  We only try the heuristics on one single dataset. It would be interesting to test them on another dataset. We could use the KBP dataset created by the natural language processing group of Stanford University[15].

[15] http://nlp.stanford.edu/

- **Event extraction on another domain:**

  We would like to try our event extraction system in other areas, including newswire corpora. For instance, we could use the knowledge base and dataset created by Reschke et al. (2014).

- **Deploy a system that extracts information about earthquakes in real time:**

  We could try extracting tweets related with the location of the earthquake on real time. This way, we could inform about earthquakes as they unfold.

- **Multilingual distant supervision:**

  Once our distant supervision system gets satisfactory results, we could make a step forward and start building the first distant supervision system for the Basque language. Resources for this language are low, compared to other languages like English. Even if corpora and knowledge bases are more limited, we could check how does the approach work with low resource languages. This could also be a great chance to explore multilingual and cross-lingual distant supervision.

4 Reading guide to the dissertation

The main contributions are published in their respective papers. We will list here these publications, organized according to the dissertation chapters:

- **Chapter 3**:
  - Ander Intxaurrondo, Oier Lopez de Lacalle and Eneko Agirre. **UBC at Slot Filling TAC-KBP 2011.** In *Proceeding of the TAC-KBP 2011 Workshop* (TAC-11). Gaithesburg, Maryland, USA, 2011.
- **Chapter 4**:
  - Ander Intxaurrondo, Mihai Surdeanu, Oier Lopez de Lacalle and Eneko Agirre. **Removing Noisy Mentions for Distant Supervision.** In *Proceedings of the XXIX Conference of Sociedad Española para el Procesamiento del Lenguaje Natural* (SEPLN-13). Madrid, Spain, 2013.
- **Chapters 5 and 6**:
  - Ander Intxaurrondo, Eneko Agirre, Oier Lopez de Lacalle and Mihai Surdeanu. **Diamonds in the rough: Event Extraction from Imperfect Microblog Data.** In *NAACL HLT 2015*. Denver, Colorado, USA. 2015.
  - Ander Intxaurrondo, Eneko Agirre and Oier Lopez de Lacalle. **Lurrikarei buruzko informazioa eskuratzen Twitter bidez. (Submitted)**

These publications can be found in the appendix of this report.

Bibliography

Babych, B. (2005). *Information Extraction Technology in Machine Translation: IE methods for improving and evaluating MT quality*. PhD thesis, University of Leeds.

Bach, N. and Badaskar, S. (2007). A Review of Relation Extraction. unpublished.

Fernandez, I. (2012). *Euskarazko Entitate-Izenak: identifikazioa, sailkapena, itzulpena eta desanbiguazioa*. PhD thesis, Euskal Herriko Unibertsitatea.

Gurrutxaga, A. (2014). *Idiomatikotasunaren karakterizazio automatikoa: izena+aditza konbinazioak*. PhD thesis, Euskal Herriko Unibertsitatea.

Ji, H. and Grishman, R. (2011). Knowledge base population: Successful approaches and challenges. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1148–1158.

Klebanov, B. B., Knight, K., and Marcu, D. (2004). Text simplification for information-seeking applications. In *On the Move to Meaningful Internet Systems, Lecture Notes in Computer Science*, pages 735–747. Springer Verlag.

Mihalcea, R. and Tarau, P. (2004). TextRank: Bringing order into texts. In *Proceedings of EMNLP-04and the 2004 Conference on Empirical Methods in Natural Language Processing*.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics*.

Miyanishi, K. and Ohkawa, T. (2013). A method of extracting sentences containing protein function information from articles by iterative learning with feature update. In *Computational Intelligence Methods for Bioinformatics and Biostatistics*, pages 81–94. Springer Berlin Heidelberg.

Pronoza, E., Yagunova, E., Volskaya, S., and Lyashin, A. (2014). Restaurant information extraction (including opinion mining elements) for the recommendation system. In *Human-Inspired Computing and Its Applications - 13th Mexican International Conference on Artificial Intelligence, MICAI 2014, Tuxtla Gutiérrez, Mexico, November 16-22, 2014. Proceedings, Part I*, pages 201–220.

Ravichandran, D. and Hovy, E. (2002). Learning surface text patterns for a question answering system. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 41–47.

Reschke, K., Jankowiak, M., Surdeanu, M., Manning, C. D., and Jurafsky, D. (2014). Event extraction using distant supervision. In *Proceedings of LREC*.

Riedel, S., Yao, L., and McCallum, A. (2010). Modeling relations and their mentions without labeled text. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD '10)*.

Surdeanu, M., Tibshirani, J., Nallapati, R., and Manning, C. D. (2012). Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing and Natural Language Learning (EMNLP-CoNLL)*.

Urizar, R. (2012). *Euskal lokuzioen tratamendu konputazionala*. PhD thesis, Euskal Herriko Unibertsitatea.

Appendix

This appendix includes a copy of the publications related to this dissertation. See Section 4 for the reading guide.

- Ander Intxaurrondo, Oier Lopez de Lacalle and Eneko Agirre. **UBC at Slot Filling TAC-KBP 2011.** In *Proceeding of the TAC-KBP 2011 Workshop* (TAC-11). Gaithesburg, Maryland, USA, 2011.

- Ander Intxaurrondo, Mihai Surdeanu, Oier Lopez de Lacalle and Eneko Agirre. **Removing Noisy Mentions for Distant Supervision.** In *Proceedings of the XXIX Conference of Sociedad Española para el Procesamiento del Lenguaje Natural* (SEPLN-13). Madrid, Spain, 2013.

- Ander Intxaurrondo, Eneko Agirre, Oier Lopez de Lacalle and Mihai Surdeanu. **Diamonds in the rough: Event Extraction from Imperfect Microblog Data.** In *NAACL HLT 2015*. Denver, Colorado, USA. 2015.

- Ander Intxaurrondo, Eneko Agirre and Oier Lopez de Lacalle. **Lurrikarei buruzko informazioa eskuratzen Twitter bidez. (Submitted)**

UBC at Slot Filling TAC-KBP 2011

Ander Intxaurrondo, Oier Lopez de Lacalle, Eneko Agirre
IXA NLP Group, University of the Basque Country, Donostia, Basque Country
{aintxaurrond001,oier.lopezdelacalle,e.agirre}@ehu.es

Abstract

This paper describes our submissions for the Slot Filling task of TAC-KBP 2011. The system takes as baseline the one we developed for the 2010 edition (Intxaurrondo et al., 2010), which is based on distant supervision. We did a straightforward implementation, trained using snippets of the document collection containing both entity and filler from the KB provided by the organizers. Our system does not use any other external knowledge source, with the exception of closed lists of words for some of the slots. We submitted three runs based on different datasets and inference options on the output of each classifiers. Ours run are below the median, but we obtained significant improvements from our last system.

1 Introduction

This paper describes our participation in the TAC-KBP 2011 Slot Filling task. Our system is a straightforward implementation of a distant supervision system (Mintz et al., 2009). To develop this system, we took the one developed for last year's edition, following the same steps as in Intxaurrondo et al. (2010) and making some improvements. The system was trained using snippets of the document collection containing both entity and filler from the KB provided by the organizers (a subset of Wikipedia infoboxes). Our system does not use any other external knowledge source, with the exception of closed lists of words for some of the slots.

The paper is structured as follows. In Section 2 the Slot Filling task will be described. In Section 3 the main components for the distant supervision system will be explained, including slot preparation, extraction of training examples, classifiers and the inference heuristics to produce the output. Next, we will focus on the results obtained by our three runs. Section 5 is devoted to error analysis, and finally, in Section 6, we draw some conclusions.

2 Slot Filling

The Slot Filling task in TAC-KBP consists on learning a set of predefined relationships and attributes for named entities (people or organizations) based on a pre-existing knowledge base extracted from Wikipedia Infoboxes. The learned information is then used to extract new facts from a large document base (1,7 million documents) for a set of target entities. The main objective is thus to feed Wikipedia Infoboxes with new additional values extracted from the document collection.

The information in the KB is organized around *entity-slot-filler* triples. An entity is the name of the article of Wikipedia, and can include people or organizations. The slot is the type of information of the entity, for example the birthplace of a person. The filler is the value of the slot. An example of an *entity-slot-filler* triple could be *Paul Newman - date of birth - January 26, 1925*. The target slots were defined by the organizers, including which are the possible fillers, and made explicit in the task guidelines.

3 Distant supervision system

In 2010, we tried a straightforward strategy for Slot Filling (Intxaurrondo et al., 2010), designed

around distant supervision (Mintz et al., 2009) and the joint work by Stanford and UBC in TAC-KBP 2009 (Agirre et al., 2009). This year we worked with the same system of 2010, and improved the results of the previous year.

Our systems has a training phase and an application (or test) phase. For training we perform the following steps:

- Slot preparation, including the extraction of entity-slot-filler triples from infoboxes, mapping them to *official* KB slots, and assigning a named-entity type or a closed list depending on the expected fillers.
- Example extraction, where we retrieve text fragments which include both the entity and filler in the triples
- Training of classifiers using the extracted examples

When applying the system we perform the following steps:

- Search of examples of mentions to the target entities
- Identification of potential fillers for possible slots
- Applying the classifiers to each filler in each mention
- Collation of results, where for each entity and slot the system returns the filler with maximum weight from classifiers[1]. When no filler is above threshold, the system returns NIL.

The development of the system did not involve manual curation of data, except assigning named entity classes (e.g., date, person) or closed lists of fillers (e.g., religions, countries,...) to each slot, as described below.

We will now present the details of how we prepared the slot information, followed by how we extracted the textual fragments (examples) of entity occurrences, and by the method to train the classifiers. The application of the classifier to produce the Slot Filling results is explained next.

[1]We tried different inference strategies in the three submissions (cf. Section 3.5.2 and 4) following this idea.

3.1 Slot Preparation

In order to prepare the training data for the slot classifiers, we first extracted entity-slot-filler triples from Wikipedia infoboxes using the mapping provided by the organizers.

As part of slot preparation, different slots based on the expected NE type were categorized (see Table 1: `ORG`, `PER`, `LOC`, `DATE`, and `NUMBER`). The NE type is used to help assign ambiguous infobox values to the appropriate slot, as well as to identify potential fillers for a text fragment for a slot. For `org:website`, regular expressions were used. Closed lists are used to improve the assignment of name-entities, they are taken from Surdeanu et al. (2010), as well as the regular expression for websites.

After obtaining the entity-slot-filler triples, we extract examples from the document base for training and development. The mapping of the infoboxes was made and provided to us by Mihai Surdeanu from Stanford University.

3.2 Train Example Extraction

The training examples were drawn from the 2010's TAC KBP Corpus. Due to time limitations we were not able to build a training set using all entity-slot-filler triples, so we used approximately 10%. We indexed the document base using the *KBP_Toolkit* search tool provided by NIST, which had Lucene on its base.

In order to extract the training examples, we used the known entity and filler pairs, and looked for occurrences of these in the document base. Exact string match is used for both the entities and fillers. We looked for examples with up to 10 tokens between the entity and filler, and five words to surrounding the entity and filler. The examples are of the form:

```
5w entity 0-10w filler 5w
5w filler 0-10w entity 5w
```

where Nw corresponds to N words/tokens; for the middle span, this ranged from zero to ten.

Note that because we look for exact matches for the entity and filler, we miss examples that contain variations of the entity or filler strings (see below).

We tried two variation. In the first we use all spans obtained for training and testing. Note that the spans

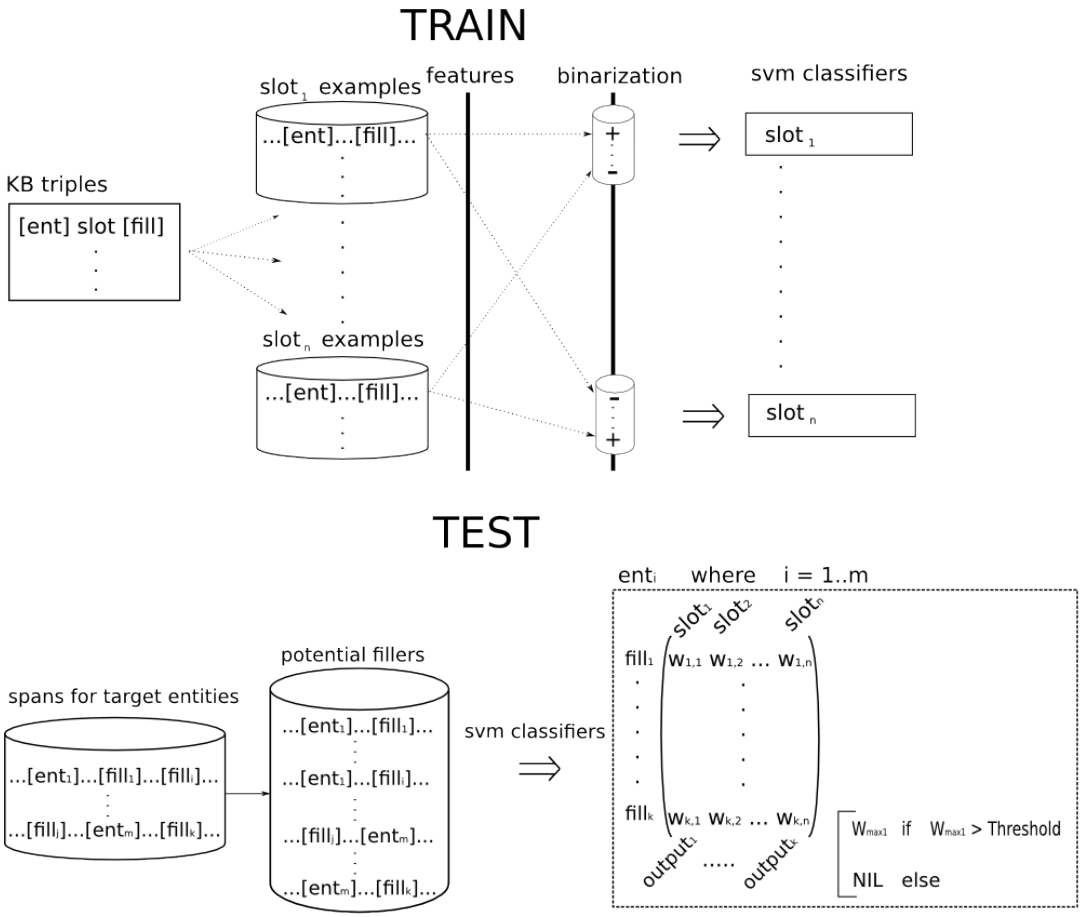

Figure 1: The architecture of the Slot Filling system. TRAIN: Extraction of KB triples, which are used to acquire training examples for each slot ($1..n$ slots) from the document base, followed by featurization and binarization. We then train n classifiers, one per slot. TEST: examples containing mentions to the target entities (m entities) are retrieved from the document base (m target entities). Potential fillers are identified, and then each example containing one entity-filler is classified, obtaining a weighted prediction for each slot. Predictions are collated and the result returned.

NE (ORG)	org:alternate_names, org:founded_by, org:member_of, org:members, org:parents, org:shareholders, org:subsidiaries, per:employee_of, per:member_of, per:schools_attended
NE (PER)	org:founded_by, org:shareholders, org:top_members/employees, per:alternate_names, per:children, per:other_family, per:parents, per:siblings, per:spouse, per:other_family, per:parents, per:siblings, per:spouse
NE (LOC)	org:city_of_headquarters, per:city_of_birth, per:city_of_death, per:cities_of_residence
NE (DATE)	org:dissolved, org:founded, per:date_of_birth, per:date_of_death
NE (NUMBER)	org:number_of_employees/members, per:age
Closed List	org:political/religious_affliation, org:country_of_headquarters, org:stateorprovince_of_headquarters, per:country_of_birth, per:country_of_death, per:countries_of_residence, per:stateorprovince_of_birth, per:stateorprovince_of_death, per:stateorprovinces_of_residence, per:cause_of_death, per:charges, per:origin, per:religion, per:title
RegExp	org:website

Table 1: Mapping of slot to NE type or closed list.

previously obtanined may contain parts of different sentences, mostly in the test set, between the entity and filler. In the second variation, we eliminate all spans containing periods (".") between the entity and filler. Each of this variations was tried in a different run, UBC1 and UBC2 respectively.

3.3 Training the Classifiers

For each slot, we trained a binary classifier that takes a text fragment with the entity and potential filler and decides whether or not the potential filler is an actual filler for the slot. We used Support Vector Machines (SVM) trained on the entity-slot-filler examples extracted from the document base (cf. Section 3.2). We deployed *svmperf*[2], which is an extension of *svmlight* to manage large sets of data, as implementation of a linear SVM classifier. Basically, our development consisted of feature set selection and setting of the SVM cost parameter (C).

For positive examples, we used examples containing the known entity and filler pairs based on slots derived from Wikipedia infoboxes. To avoid misleading infoboxes, we only used examples that had an entity type matching the entity type of the slot.

For negative examples, we distinguish between persons and organizations. For instance, given a specific classifier of slot i for person entity, the rest of the person slots were considered as negative examples. We followed the same strategy for slots of organization entities.

Regarding learning features, in related experiments on the ACE 2005 dataset we carried out a selection of the learning features. Our system make use of the features introduced by Mintz et al. (2009), the ones proposed by Zhou et al. (2005), and some of the surface features proposed by Surdeanu et al. (2010).

This way, following Mintz et al. (2009) we extracted the following feature types:

- The sequence of words between the entity and filler (10 words maximum).
- The part-of-speech tags of these words.
- The name-entity types of the entity and filler.
- A window of *k* words to the left of the first entity/filler and their part-of-speech tags
- A window of *k* words to the right of the second entity/filler and their part-of-speech tags.

Each lexical feature consists of a conjunction of all this components. We generate a conjunctive feature for each $k \in \{0, 1, 2\}$.

Features based on Zhou et al. (2005) are the following types:

- A flag indicating there is no word between the entity and filler.
- A flag indicating there is only one word between the entity and filler.
- The first word after the first-coming entity/filler.
- The last word before the second-coming entity/filler.
- All words between the entity and filler, except the first and last.
- The first word before the first-coming entity/filler.
- The second word before the first-coming entity/filler.
- The first word after the second-coming entity/filler.
- The second word after the second-coming entity/filler.
- The name-entities of the entity and filler.

And finally, features based on Surdeanu et al. (2010) are the following ones, with some extras:

- A flag indicating if the entity comes before the filler or the filler before the entity.
- Distance between entity and filler.
- A flag indicating the word form of the entity. If the entity is formed by more than one word, all these words are separated by "_"
- Some flags indicating all words in the entity separately. If the entity is formed by just one word, there will only be one flag.
- A flag indicating the word form of the filler. If the filler is formed by more than one word, all these words are separated by "_"
- Some flags indicating all words in the filler separately. If the filler is formed by just one word, there will only be one flag.
- Entity's part-of-speech.
- Filler's part-of-speech.
- Entity's name-entity type.
- Filler's name-entity type.

Table 2 shows the resulting lexical feature (note

[2] http://www.cs.cornell.edu/People/tj/svm_light/svm_perf.html

that the each row in the table represents a single lexical feature).

3.3.1 Optimizing C

Due to the importance of the C parameter in SVM classifiers tried values of C ranging from 0.01 to 20 in the 2010 Slot Filling dataset. This way, we learnt the best C value for each of the submitted run, as shown in Table 4. for each run.

3.4 Getting test examples

In order to get examples where potential filler could be found for the target entities, we extracted examples in the document base that matched the string of the target entity exactly. These examples are of the form:

```
30w entity 30w
```

We also wanted to test whether examples of the variants of the target entity, as listed in Wikipedia, would increase the performance of the system. We used these additional test examples for the `UBC3` run.

3.5 Applying the classifiers

Once the classifiers were trained, we used them to determine the most likely fillers for the target entities. Using the examples extracted from the document base for each entity, we identified potential fillers using a NER module or closed lists of strings (see Table 1). After identifying potential fillers within the span, we expanded the examples for target entities in entity-filler pairs (see Figure 1, test part). For each entity-filler pair extraction of features was carried out, and the prediction of the classifier in the slot was obtained deciding whether the filler was positive or negative.

3.5.1 Optimizing the threshold

We learnt the optimum threshold to decide if a potential filler could be considered as a candidate filler. As we did with the C parameter with the classifier, for each best C we tried different threshold values of the classifiers predictions. We optimized the system according the threshold values between -1 and 1. The chosen parameter values where the ones that gave the best results with the target-entities used in the 2010 Slot Filling task.

If a slot had all the filler predictions below the threshold, the system would return a NIL value for that slot (see Figure 1, "Output" part).

3.5.2 Inference

Rather than returning the maximum filler directly, we first checked if the top-scoring filler was compatible with the slot; if the filler's name-entity type was compatible, we considered the potential filler as positive; if not, then we rejected that filler and checked if the next top-scoring filler was compatible or not; and so on until we found one. As an example, lets suppose that we are checking potential fillers for slot *per:date_of_birth*, the correct filler should be a date, but if we obtain as top-scoring filler a person's name, then we reject it and check the next one. This is the strategy used in our `UBC1` and `UBC2` runs.

As an additional piece of evidence, we also considered the frequency of a potential filler for each target entity. After checking the compatibility of each prediction, we take each potential filler and sum its prediction, even if that prediction is negative and below threshold in that slot. Once we obtain each ponential filler's sum, the take the top 3 sums of slots for that potential filler, check if every sum is above threshold for a slot, and if it is, consider that slot as a relation for the target entity and that potential filler. This was used in the `UBC3` run.

3.6 Improvements from TACKBP Slot Filling 2010 to 2011

The improvements of our Information Extraction system from the system developed in 2010 to the one developed in 2011 are the following:

- Better dataset: The *entity-slot-filler* triplets where less noisier in 2011. This constructed a cleaner dataset.
- Synonyms: Test sets were increased with more span examples. These extra examples contain synonyms of the target entity.
- Learning features: We learned last year that we needed to develop a supervised IE system before jumping to distant supervision. During 2011, we worked with the ACE 2005 corpus, using the same features as in 2010 from the beginning, to improve them, add more features,

ENTITY - SLOT - FILLER : Dominican University - org:city_of_headquarters - River Forest
SPAN: *...courses at <entity> Dominican University </entity> in the Chicago suburb of <filler> River Forest </filler> shortly before...*

LEFT WINDOW	NE1	MIDDLE	NE2	RIGHT WINDOW
[]	ORGANIZATION/ENTITY	in/IN the/DT Chicago/NN suburb/NN of/IN	LOCATION/FILLER	[]
[at/IN]	ORGANIZATION/ENTITY	in/IN the/DT Chicago/NN suburb/NN of/IN	LOCATION/FILLER	[]
[courses/NN at/IN]	ORGANIZATION/ENTITY	in/IN the/DT Chicago/NN suburb/NN of/IN	LOCATION/FILLER	[]
[courses/NN at/IN]	ORGANIZATION/ENTITY	in/IN the/DT Chicago/NN suburb/NN of/IN	LOCATION/FILLER	[shortly/RB]
[courses/NN at/IN]	ORGANIZATION/ENTITY	in/IN the/DT Chicago/NN suburb/NN of/IN	LOCATION/FILLER	[shortly/RB before/IN]
[at/IN]	ORGANIZATION/ENTITY	in/IN the/DT Chicago/NN suburb/NN of/IN	LOCATION/FILLER	[shortly/RB before/IN]
[at/IN]	ORGANIZATION/ENTITY	in/IN the/DT Chicago/NN suburb/NN of/IN	LOCATION/FILLER	[shortly/RB]
[]	ORGANIZATION/ENTITY	in/IN the/DT Chicago/NN suburb/NN of/IN	LOCATION/FILLER	[shortly/RB]
[]	ORGANIZATION/ENTITY	in/IN the/DT Chicago/NN suburb/NN of/IN	LOCATION/FILLER	[shortly/RB before/IN]

FEATURE TYPE	VALUE
WBF	in
WBL	of
WBO	the Chicago suburb
BM1F	courses
BM1L	at
AM2F	shortly
AM2L	before
DIR	ENTFILL
DIST	5
ENT	Dominican_University
ENT1	Dominican
ENT1	University
FILL	River_Forest
FILL1	River
FILL1	Forest
ENTPOS	NN
FILLPOS	NN
ENTTYPE	ORGANIZATION
FILLTYPE	LOCATION

Table 2: Example of the features used. The first 9 lines represent Mintz et al., while the next 7 lines correspond to Zhou et al., the last ones are from Surdeanu et al.

and analyze which features fitted better. The features that gave the best performance are used here.

- Optimization: Working with ACE, we also used SVM, and optimized the system using different C values with 5-fold cross-validation. Due that the final answers improved a lot with different values, we used the same technique in TACKBP 2011. SVM gives numerical predictions for each potential filler, using different thresholds to consider a filler as valid gives even better answers.

4 Results

The core of our system is the same used at the 2010 Slot Filling task (Intxaurrondo et al., 2010). We submitted three runs based on different datasets and post-processing of the output of the classifiers.

For the first run (UBC1), we used all the *entity-slot-filler* spans obtained as training dataset. Meanwhile, for the second (UBC2) and third (UBC3) run, we removed spans that contained periods between the entity and filler in the training set. The test set in the third run contains extra spans searched using synonyms of the target entity.

For the first and second run, we control if the top-scoring potential fillers for each slots and the slot type are compatible, checking their name-entity types (cf. Section 3.4.2). In the third run, apart of checking for compatibility, we sum their prediction values, and if their sum if above the threshold value, we check the top three slots where they have the maximun value.

Table 3 shows the values of the C parameter and prediction threshold used for each run.

Table 4 shows the official results in TAC 2011 KBP Slot Filling task, followed by the median. The second run shows an improvement when using spans with the entity and filler are in the same sentence.

	SVM - C value	Prediction Threshold
Run 1	5	-0.5
Run 2	1	-0.5
Run 3	5	-0.25

Table 3: Parameters used for each run.

	UBC1	UBC2	UBC3	median
Recall	2.96	2.85	3.28	10.31
Precision	4.45	5.36	4.74	16.50
F1	3.55	3.72	3.87	12.69

Table 4: TAC 2011 KBP Slot Filling Results.

We obtain the best results with the third run, adding spans with synonyms for the target entity and using frequency information.

4.1 What did not work

We will shortly review some of the techniques which did not work:

- Trying to remove noisy examples from the dataset, we took all *entity-slot-filler* examples and randomly took only one example per triplet. The f-measure decreased.
- Hoping to obtain better results, we combined all relations tagged in the ACE corpora with the Slot Filling task relations with the help of the annotation guidelines[3]. Due that ACE examples are tagged by hand and its lack of noise, we expected to increase the f-measure, but the final results were worse.

We also tried different the following heuristics for inference:

- For each slot, sum prediction values to each potential filler, and get the maximun filler as correct in case it was higher than the threshold value.
- For each potential filler, give as answer top three slots where they have highest prediction values only if the prediction values are higher than the threshold value.
- For each slot, we selected all potential fillers mentioned more than once, and then we calculated the average prediction value. We selected the potential filler with the maximun average value only if the average value was higher than the threshold value. Finally, if the selected filler was compatible with the slot, we considered it as valid.

None of them gave good results comparing to the combined heuristic used in run 3 (UBC3).

5 Analysis

Our system developed for the TACKBP 2011 is a significant improvement compared to the one developed for TACKBP 2010 ((Intxaurrondo et al., 2010)), but it's still weak, due to the following reasons:

- **Noisy positive examples**. Although this time the system had less noisy examples generated from the beginning, many of the gathered training examples were still inaccurate for appropriate automatic learning. This means that we should apply some kind of filtering or instance weighting technique to get rid of useless examples.
- **Lack of positive examples**. There were some slots with no positive examples in the training set, such as *per:charges, per:children, per:other_family* and *org:shareholders*.
- **Negative examples**. We generated too many negative example producing an unbalanced training set. Unbalanced training sets introduce undesirable biases in the learning process. Smart filtering of negative examples or weighted SVM classifiers might be a desirable solution to the problem. In Table 5 we show the number of tuples, positive spans and negative spans for the slots; note the excess of positive examples por slot *per:country_of_birth*, this slot makes person slots to be very umbalanced, fortunately this does not happen in organization slots. Slots like *per:cause_of_death* have no positive examples, having the maximun number of negatives.

6 Conclusions

We have participated with a preliminary implementation of a distant supervision system. The idea

[3] http://projects.ldc.upenn.edu/ace/docs/English-Relations-Guidelines_v5.8.3.pdf

slot	triples	pos. examples	neg. examples	slot	triples	pos. examples	neg. examples
per:age	81	152	43098	org:alternate_names	71	483	11344
per:alternate_names	41	522	42728	org:city_of_headquarters	214	2068	9759
per:cause_of_death	0	0	43250	org:country_of_headquarters	165	1235	10592
per:charges	0	0	43250	org:dissolved	93	335	11492
per:children	249	1943	41307	org:founded	32	103	11724
per:cities_of_residence	120	269	42981	org:founded_by	96	313	11514
per:city_of_birth	77	359	42891	org:member_of	47	176	11651
per:city_of_death	66	371	42879	org:members	155	1721	10106
per:countries_of_residence	252	986	42264	org:number_of_employees/members	37	127	11700
per:country_of_birth	2343	22593	20657	org:parents	70	370	11457
per:country_of_death	103	784	42466	org:political/religious_affiliation	189	2136	9691
per:date_of_birth	113	200	43050	org:shareholders	0	0	11827
per:date_of_death	7	8	43242	org:stateorprovince_of_headquarters	166	1190	10637
per:employee_of	171	2289	40961	org:subsidiaries	79	1434	10393
per:member_of	153	1018	42232	org:top_members/employees	49	127	11700
per:origin	235	1194	42056	org:website	6	8	11819
per:other_family	0	0	43250				
per:parents	90	1144	42106				
per:religion	94	564	42686				
per:schools_attended	64	145	43105				
per:siblings	5	10	43240				
per:spouse	89	237	43013				
per:stateorprovince_of_birth	75	268	42982				
per:stateorprovince_of_death	197	544	42706				
per:stateorprovinces_of_residence	474	7066	36184				
per:title	103	579	42617				

Table 5: Statistics for all slots, including number of triples, positive and negative examples used in UBC2 and UBC3.

was to train the system using snippets of the document collection containing both entity and filler from the KB provided by the organizers (a subset of Wikipedia infoboxes). Our system does not use any other external knowledge source, with the exception of closed lists of words for religion, causes of death, charges and religious/political affiliation, and many more.

We submitted three runs, with different training and testing example settings, based on different post-processing options of the output of our classifiers. We have seen that using synonyms of the target entities improves the recall, and that inference can be used to filter wrong fillers.

Our main goal was to improve over last year's system, which we accomplished, but are still below the median. For the future we plan to focus on methods to deal with the noise in the examples.

Acknowledgements

We would like to thank Stanford researchers Mihai Surdeanu and Julie Tibshirani for providing us the Knowledge Base mapped to slots. Ander Intxaurrondo has a grant from the University of the Basque Country. Part of this research is funded by the European Commission (PATHS ICT-2011-270082) and the Ministry of Science and Innovation (KNOW2 TIN2009-14715-C04-01).

References

Eneko Agirre, Angel X. Chang, Daniel S. Jurafsky, Christopher D. Manning, Valentin I. Spitkovsky, and Eric Yeh. 2009. Stanford-UBC at TAC-KBP. In *Proceedings of the Second Text Analysis Conference (TAC 2009)*, Gaithersburg, Maryland, USA, November.

Ander Intxaurrondo, Oier Lopez de Lacalle, and Eneko Agirre. 2010. UBC at Slot Filling TAC-KBP 2010. In *Proceedings of the Third Text Analysis Conference (TAC 2010)"*, Gaithersburg, Maryland, USA, November.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of ACL-IJCNLP 2009*.

Mihai Surdeanu, David McClosky, Julie Tibshirani, John Bauer, Angel X. Chang, Valentin I. Spitkovsky, and Christopher D. Manning. 2010. A Simple Distant Supervision Approach for the TAC-KBP Slot Filling Task. In *Proceedings of the TAC-KBP 2010 Workshop*.

GuoDong Zhou, Jian Su, Jie Zhang, and Min Zhang. 2005. Exploring various knowledge in relation extraction. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 427–434, Morristown, NJ, USA. Association for Computational Linguistics.

Removing Noisy Mentions for Distant Supervision

Eliminando Menciones Ruidosas para la Supervisión a Distancia

Ander Intxaurrondo*, Mihai Surdeanu, Oier Lopez de Lacalle***, Eneko Agirre***
*University of the Basque Country. Donostia, 20018, Basque Country
**University of Arizona. Tucson, AZ 85721, USA
***University of Edinburgh. Edinburgh, EH8 9LE, UK
ander.intxaurrondo@ehu.es, msurdeanu@email.arizona.edu,
oier.lopezdelacalle@ehu.es, e.agirre@ehu.es

Resumen: Los métodos para Extracción de Información basados en la Supervisión a Distancia se basan en usar tuplas correctas para adquirir menciones de esas tuplas, y así entrenar un sistema tradicional de extracción de información supervisado. Un problema de la supervisión a distancia es el ruido introducido en las menciones extraidas. En este artículo analizamos las fuentes de ruido en las menciones y exploramos métodos simples para filtrar menciones ruidosas. Los resultados demuestran que combinando el filtrado de tuplas por frecuencia, la información mutua y la eliminación de menciones lejos de los centroides respectivos mejora los resultados de dos métodos de extracción de información significativamente.
Palabras clave: Extracción de Información, Extracción de Relaciones, Supervisión a Distancia, Aprendizaje con Ruido

Abstract: Relation Extraction methods based on Distant Supervision rely on true tuples to retrieve noisy mentions, which are then used to train traditional supervised relation extraction methods. In this paper we analyze the sources of noise in the mentions, and explore simple methods to filter out noisy mentions. The results show that a combination of mention frequency cut-off, Pointwise Mutual Information and removal of mentions which are far from the feature centroids of relation labels is able to significantly improve the results of two relation extraction methods.
Keywords: Information Extraction, Relation Extraction, Distant Supervision, Learning with Noise

1 Introduction

Distant Supervision (DS) is a semi-supervised alternative to traditional Relation Extraction (RE) that combines some of the advantages of different RE approaches. The intuition is that any sentence that contains a pair of entities that are recorded in a Knowledge Base (KB) such as DBpedia[1] or Freebase[2] to participate in a known relation (e.g., *born-in* or *film-director-of*) is likely to provide evidence for that relation. Using this approach, large training datasets of relation mentions can be automatically created by aligning entities that participate in known relations with sentences from large corpora where the entity pairs are mentioned. Such sentences are preprocessed to identify all named or numeric entities that are mentioned. Entities are identified using named entity recognizers, tagging them as persons, organizations, locations, dates, etc. If the KB specifies that a pair of entities appearing in the same sentence participates in a known relation, the corresponding textual context becomes a mention for the corresponding relation label. If the KB has no record of the two entities, the corresponding relation is marked as *unrelated* (i.e., a negative mention). Using this approach, a very large number of relation mentions can be gathered automatically, thus alleviating the sparse data problem plaguing supervised relation extraction systems, which ultimately causes overfitting and domain dependence.

In order to illustrate the method, let's consider some relations[3] and tuples from Free-

[1] http://dbpedia.org/About
[2] http://www.freebase.com/

[3] In order to improve readability, we will use intuitive tags instead of the actual Freebase relation names, i.e., *education* for */education/education/student*, *capital* for */lo-*

base:

- <Albert Einstein, *education*, University of Zurich>
- <Austria, *capital*, Vienna>
- <Steven Spielberg, *director-of*, Saving Private Ryan>

Searching for the entity pairs in those tuples, we can retrieve sentences that express those relations:

- **Albert Einstein** was awarded a PhD by the **University of Zurich**.
- **Vienna**, the capital of **Austria**.
- Allison co-produced the Academy Award-winning **Saving Private Ryan**, directed by **Steven Spielberg**...

Although we show three sentences that do express the relations in the knowledge-base, distant supervision generates many noisy mentions that hurt the performance of the relation extraction system. We identified three different types of noise in the mentions gathered by distant supervision:

1. Sentences containing related entities, but which are tagged as 'unrelated' by DS. This happens because the KB we use, as all real-world Kbs, is incomplete, i.e., it does not contain all entities that participate in a given relation type.
2. Sentences containing unrelated entities, tagged as related. This happens when both participating entities that are marked as related in the KB appear in the same sentence, but the sentence does not support the relation.
3. Sentences containing a pair of related entities, but which are tagged as a mention of another, incorrect, relation. This type is the most common, and happens for entity tuples that have more than one relational label. These were previously called multi-label relations in the literature (Hoffmann et al., 2011).

Suppose that we have an incomplete KB according to whom the tuple <Brazil, Celso Amorim> is unrelated. In reality Celso is a minister of Brazil, and thus a mention of the *country-minister* relation. Mentions like *Celso Amorim , the Brazilian foreign minister , said the (...)* will be tagged by DS systems as unrelated at the training dataset, instead of appearing as *country-minister* as it should be. This is an example of Type 1.

Situations of Type 2 noise occur with tuples like <Jerrold Nadler, *born_in*, Brooklyn>. If the system extracts the following sentences from the corpora, *(...) Representative Jerrold Nadler, a Democrat who represents parts of Manhattan and Brooklyn, (...)* and *Nadler was born in Brooklyn, New York City.*, they both will be tagged as *born_in* and used later for training, although the entity tuple in the first sentence is not a positive mention of the relation under consideration.

Below we give an example of Type 3 noise. Consider the tuple <Rupert Murdock, News Corporation>. This is a multi-label relation with labels *founder* and *top-member*. Thus sentences in the training set such as *News Corporation was founded by Rupert Murdock* and *Rupert Murdock is the CEO of News Corporation* will be both considered as mentions for both *founder* and *top-member*, even though the first sentence is not a mention for the *top-member* relation and the second is not a mention for the *founder* relation.

We selected randomly 100 mentions respectively from single-label related mentions, multi-labeled related mentions and unrelated mentions which correspond to Freebase relations as gathered by (Riedel, Yao, and McCallum, 2010). We analyzed them, and estimated that around 11% of the unrelated mentions belong to Type 1, 28% of related single-labeled mentions belong to Type 2. Regarding multi-labeled mentions, 15% belong to Type 3 and 60% to Type 2, so only 25% are correct mentions. All in all, the dataset contains 91373 unrelated mentions, 2330 single-labeled and 26587 multi-labeled mentions, yielding an estimate of 29% correct instances for related mentions, and 74% correct instances overall.

Noisy mentions decrease the performance of distant supervision systems. However, because the underlying datasets are generally very large, detecting and removing noisy mentions manually becomes untenable. This paper explores several methods that automatically detect and remove noisy mentions generated through DS.

cation/country/capital, and *director-of* for */film/director/film*

2 Related Work

Distant Supervision was originally proposed by (Craven and Kumlien, 1999) for the biomedical domain, extracting binary relations between proteins, cells, diseases and more. Some years later, the approach was improved by (Mintz et al., 2009), making it available for different domains, such as *people, locations, organizations,...*, gaining popularity since then.

We can find many approaches that model the noise to help the classifier train on the respective datasets. (Riedel, Yao, and McCallum, 2010) model distant supervision for relation extraction as a multi-instance single-label problem, allowing multiple mentions for the same tuple, but it does not allow more than one label per object. (Hoffmann et al., 2011) and (Surdeanu et al., 2012) focus on multi-instance multi-label learning.

Distant supervision has also been the most relevant approach used to develop different relation extraction system at the *TAC-KBP Slot Filling* task[4] for the last years, organized by NIST. Nearly all the participants use distant supervision for their systems to extract relations for *people* and *organization* entities. The approach has improved slowly during the latest years, and working with noisy mentions to train the systems has been recognized as the most important hurdle for further improvements.

3 Distant Supervision for Relation Extraction

The methods proposed here for cleaning the textual evidence used to train a RE model are system independent. That is, they apply to any RE approach that follows the "traditional" distant supervision heuristic of aligning database tuples with text for training. As proof of concept, in this paper we use two variants of the *Mintz++* system proposed by (Surdeanu et al., 2012) and freely available at `http://nlp.stanford.edu/software/mimlre.shtml`. This algorithm is an extension of the original work of (Mintz et al., 2009) along the following lines:

- The *Mintz++* approach models each relation mention independently, whereas Mintz et al. collapsed all the mentions of the same entity tuple into a single datum. In other words, *Mintz++* constructs a *separate* classification data point from every sentence that contains a training tuple, whereas the original Mintz et al. algorithm merges the features extracted from all sentences that contain the same tuple into a single classification mention. The former approach was reported to perform better in practice by (Surdeanu et al., 2012).
- *Mintz++* allows multiple labels to be predicted for the same tuple by performing a union of all the labels proposed for individual mentions of the same tuple, whereas the Mintz et al. algorithm selected only the top-scoring label for a given entity pair. The multiple-label strategy was also adopted by other models ((Hoffmann et al., 2011); (Surdeanu et al., 2012)). This is necessary because the same pair of entities may express multiple relations, e.g., (*Balzac*, *France*) expresses at least two relations: *BornIn* and *Died*, which cannot be modeled by Mintz et al.'s algoritm.
- *Mintz++* implements a bagging strategy that combine five individual models. Each model is trained using four out of five folds of the training corpus. The final score is an unweighted average of the five individual scores. In this paper, we report results using two variants of the *Mintz++* model: when this ensemble modeling strategy is enabled (*Mintz++*) or disabled, i.e., using a single model trained over the entire training data (which we will call *Mintz**). This allows us to directly compare the effects of bagging with the impact of the data-cleanup proposed in this paper.

The results reported here are generated over the corpus created by (Riedel, Yao, and McCallum, 2010) and used by many other IE researchers like (Hoffmann et al., 2011), (Surdeanu et al., 2012), inter alia. This corpus uses Freebase as the source for distant supervision and the New York Times (NYT) corpus by (Sandhaus, 2008) for the source of textual evidence. The corpus contains two partitions: a training partition, containing 4700 relation mentions from the 2005–2006 portion of the NYT corpus, and a testing

[4]Task definition for 2013 available at `http://surdeanu.info/kbp2013/KBP2013_TaskDefinition_EnglishSlotFilling_1.0.pdf`

partition, containing 1950 more recent (2007) relation mentions. Because this corpus does not have a reserved development partition, we tuned our models over the training partition using cross-validation. In both partitions, negative mentions were automatically generated from pairs of entities that co-occur in the same sentence and are not recorded in Freebase with any relation label. Crucially, the corpus authors released only a random subsample of 5% of these negative mentions for the training partition. This means that any results measured over the training partition will be artificially inflated because the systems have fewer chances of inferring false positive labels.

4 Methods to Remove Noise

We tried three different heuristics to clean noisy mentions from the dataset. We experimented removing tuples depending on their mention frequency (MF), their pointwise mutual information (PMI), and the similarity between the centroids of all relation mentions and each individual mention (MC). We also built several ensemble strategies that combine the most successful individual strategies, as parametrized over development data. Note that none of these methods uses any additional manual annotation at all.

4.1 Mention Frequency (MF)

For our first heuristic, we consider that tuples with too many mentions are the most probable to contain noisy mentions, so we remove all those tuples that have more than a predefined number of mentions. Our system removes both positive tuples that appear in Freebase, and negative (unrelated) tuples which contain more than X mentions. We experimented with different thresholds and chose the limit that gave the highest F-Measure on the development set[5]. The chosen value was $X = 90$, i.e., all tuples with more than 90 mentions where removed, around 40% of the positive mentions, and 15% of the total dataset considering both positive and negative mentions.

For example, the tuple <European Union, *has-location*[6], Brussels> appears with 95 mentions. This tuple contains good mentions like *The European Union is headquartered in Brussels.* but also many noisy mentions like *The European Union foreign policy chief, Javier Solana, said Monday in Brussels that (...)* or *At an emergency European Union meeting of interior and justice ministers in Brussels on Wednesday, (...)* which do not explicitly say that Brussels is in the European Union, and can thus mislead the supervised RE system. This heuristic removes all instances of this tuple from the training data.

4.2 Tuple PMI

The second heuristic calculates the PMI between each entity tuple and the a relation label. Once we calculate the PMI for each tuple, we consider that the tuples which have a PMI below a defined threshold have noisy mentions, and remove them. Empirically, we observed that our system performs better if we remove only positive mentions with low PMI and keep the negative ones, regardless of their PMI value. Our system performed better with a threshold of 2.3, removing around 8% of the positive training tuples. This heuristic is inspired by the work of (Min et al., 2012).

As an example, this approach removed the tuple: <Natasha Richardson, *place-of-death*[7], Manhattan>. This tuple has only one mention: *(...) Natasha Richardson will read from 'Anna Christie,' (...) at a dinner at the Yale Club in Manhattan on Monday night.*. This mention does not support the place-of-death relation. That is, even though Natasha Richardson died in Manhattan, the mention is unrelated to that fact.

4.3 Mention Centroids (MC)

This heuristic calculates the centroid of all mentions with the same relation label, and keeps the most similar mentions to the centroids. We hypothesize that the noisy mentions are the furthest ones from their label centroids. For this experiment, we consider each mention as a vector and the features as space dimensions. We use the same features used by the DS system to build the vectors, with the frequency as the value of the feature. The centroid is built from the vectors as described in equation 1 below.

$$\vec{C_i} = \left(\frac{feat_1}{mentions_i}, \frac{feat_2}{mentions_i}, \ldots, \frac{feat_N}{mentions_i} \right) \quad (1)$$

[5] Throughout the paper, development experiments stand for cross-validation experiments on Riedel's training partition.

[6] /location/location/contains

[7] /people/deceased_person/place_of_death

where mentions_i = number of mentions for label i ($1 \leq i \leq M$), feat_j = number of appearances of feature j ($1 \leq j \leq N$) and C_i = Centroid for label i.

The similarity between a centroid and any given mention is calculated using the cosine:

$$cosine(C, M) = \frac{\vec{C} \cdot \vec{M}}{\sqrt{\vec{C} \cdot \vec{C}} \cdot \sqrt{\vec{M} \cdot \vec{M}}} \quad (2)$$

where C = Centroid and M = Mention.

We select a percentage of the most similar mentions to each centroid, and discard the rest. Our system returned the best results on development when we kept 90% of the most similar mentions of each relational label.

We do not use this heuristic for negative mentions. Empirically, we observed that this heuristic performs better if we kept all negative mentions rather than deleting any of them. This could be an artifact of the fact that only 5% of the negative mentions are included in Riedel's training dataset. Thus, sub-sampling negative mentions further yields datasets with too few negative mentions to train a discriminative model. This methods removes around 8% of the positive mentions.

As an example of the method, if we take the centroid for relation *company-founders*, the mention appearing in the sentence *(...) its majority shareholder is Steve Case, the founder of AOL* of the tuple <Steve Case, company-founders, AOL> is the most similar to the centroid of the same label. On the contrary, the mention *Ms. Tsien and Mr. Williams were chosen after a competition that began with 24 teams of architects and was narrowed to two finalists, Thom Mayne's Morphosis being the other* of the tuple <Thom Mayne, Morphosis> was correctly excluded, as the mention does not explicitly say that Thom Mayne is the founder of Morphosis.

4.4 Ensemble Models

We experimented with several ensemble models that combine the above individual strategies, in different order. The best results on development, as shown in Section 5.1, were different for *Mintz** and *Mintz++*. For the first, we first filtered using PMI, then run the MF filter, and finally applied the centroid-based filter. For the second, the best combination was to run PMI and then MF. The MC method did not provide any additional gain.

	Rec.	Prec.	F1
Mintz*	34.98	39.44	37.07
MF 90	33.19	44.49	**38.01**
PMI 2.3	34.49	40.64	37.31
MC 90%	34.81	40.31	37.33
PMI+MF+MC	32.72	46.36	**38.53**

Table 1: Development experiments using Mintz*, showing the results of each filtering method and the best combination.

	Rec.	Prec.	F1
Mintz++	34.85	41.45	37.86
MF 180	33.65	44.48	**38.45**
PMI 2.4	34.00	42.97	37.95
PMI+MF	33.25	45.57	**38.45**

Table 2: Development experiments using Mintz++, showing the results of each filtering method and the best combination.

5 Experiments

We evaluated the methods introduced in the previous section with the dataset developed by (Riedel, Yao, and McCallum, 2010). This dataset was created by aligning Freebase relations with the New York Times (NYT) corpus. They used the Stanford named entity recognizer to find entity mentions in text and constructed relation mentions only between entity mentions in the same sentences. We used the same features as (Riedel, Yao, and McCallum, 2010) for the mention classifier.

The development set was created using a three-fold cross-validation technique, similarly to (Surdeanu et al., 2012). For the formal evaluation on the test set, we only used the best ensemble models, instead of applying each method individually.

5.1 Results on the Development Corpus

The initial experiments where done using the *Mintz++* system in (Surdeanu et al., 2012) without any ensemble at the classifier. From now on, the *Mintz++* without the ensemble will be denoted as *Mintz** in this paper. Table 1 shows the results we obtained with each method. If we execute our methods individually, we get the best results with the *Mention frequency* experiment (Section 4.1), where

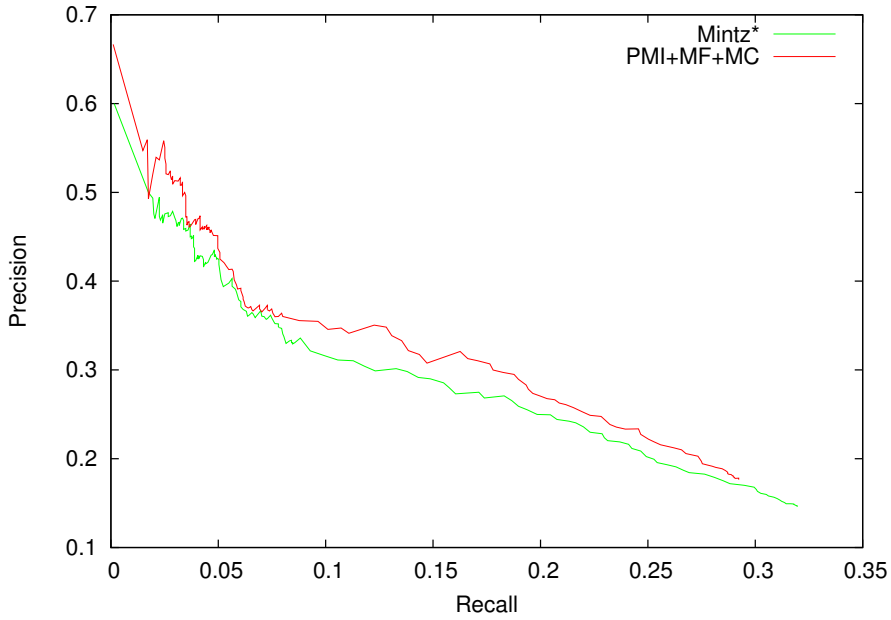

Figure 1: Precision/recall curves for the Mintz* system on the test partition. The red line is our best filtering model.

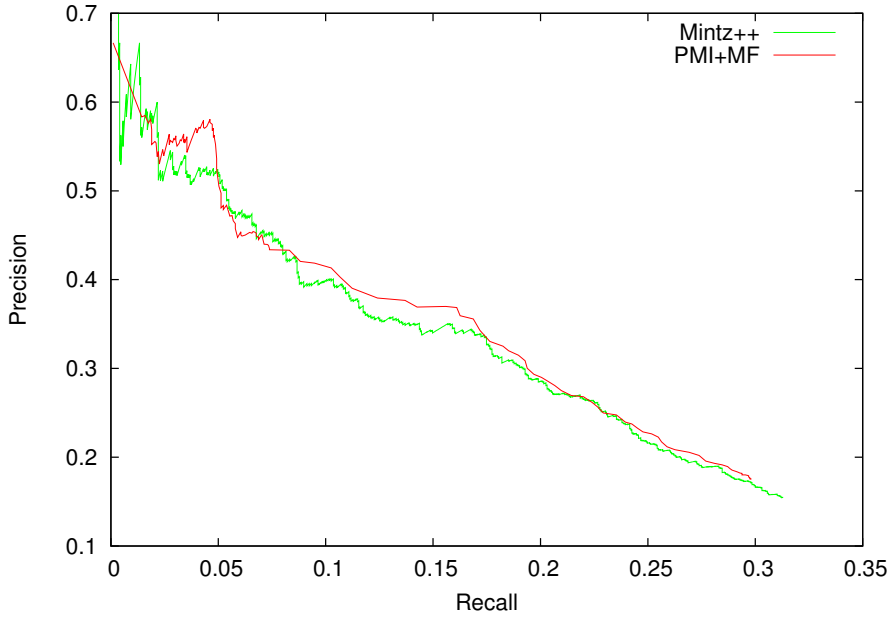

Figure 2: Precision/recall curves for the Mintz++ system on the test partition. The red line is our best filtering model.

our system's F-Measure improves nearly 1%. The PMI (Section 4.2) and the *Mention centroids* models (Section 4.3) both yield a small improvement over the baseline. For the ensemble models, we obtain the best performance by combining PMI with *Mention frequency* and the *Mention centroids*, improving the F-Measure nearly 1.5 absolute points. Our system improves the precision in each experiment, but not the recall, this scoring

	Rec.	Prec.	F1
Mintz*	31.95	14.57	20.02
PMI+MF+MC	29.23	17.64	**22.00**

Table 3: Results on the test partition for Mintz* (without bagging).

	Rec.	Prec.	F1
Mintz++	31.28	15.43	20.67
PMI+MF	29.79	17.48	**22.03**

Table 4: Results for Mintz++ (with bagging).

parameter generally decreases slightly. This is to be expected, since the models built using filtered data train on fewer positive mentions, thus they will be more conservative in predicting relation labels.

We applied the same heuristics to the original *Mintz++* system at (Surdeanu et al., 2012), and optimized them. The optimal parameters are 180 mention maximum for *Mention frequency* (4.1), and 2.4 for the *PMI* heuristics (Section 4.2). Unfortunately the *Mention centroids* (Section 4.3) heuristic did not yield an improvement here. Finally, we combined the *PMI* heuristic with the *Mention frequency* experiment to improve our results. Table 2 shows the results we obtained for each heuristic. Surprisingly, *MF 180* and *PMI+MF* give the same F-Measure.

5.2 Results on the Test Partition

For the formal evaluation on the test set, we only chose the ensemble models that performed best with the development set for *Mintz**, with the same optimal parameters obtained on development. On the test set, the F-Measure improves approximately 2 points. The results are shown in Table 3.

Figure 1 shows the precision/recall curves for our best system relative to the *Mintz** baseline. The figure shows that our approach clearly performs better.

Table 4 shows the results on the test partition of the original *Mintz++* system of (Surdeanu et al., 2012) and the *Mintz++* extended with our best ensemble filtering model (tuned on development).

Figure 2 shows the precision/recall curves of the two systems based on *Mintz++*. The models trained using filtered data perform generally better than the original system, but the differences are not as large as for the previous model that does not rely on ensemble strategies. This suggests that ensemble models, such as the bagging strategy implemented in *Mintz++*, are able to recover from some of the noise introduced by DS. However, bagging strategies are considerably more expensive to implement than our simple algorithms, which filter the data in a single pass over the corpus.

To check for statistical significance, we used the bootstrapping method proposed by (Berg-Kirkpatrick, Burkett, and Klein, 2012) verifying if the improvement provided by mention filtering is significant[8]. This bootstrapping method concluded that, although the difference between the two models is small, it is statistically significant with p-values below 0.001, thus supporting our hypothesis that data cleanup for DS algorithms is important.

6 Conclusions

Motivated by the observation that relation extraction systems based on the distant supervision approach are exposed to data that includes a considerable amount of noise, this paper presents several simple yet robust methods to remove noisy data from automatically generated datasets. These methods do not use any manual annotation at the datasets. Our methods are based on limiting the mention frequency for each tuple, calculating the Pointwise Mutual Information between tuples and relation labels, and comparing mention vectors against the mention centroids of each relation label.

We show that these heuristics, especially when combined using simple ensemble approaches, outperform significantly two strong baselines. The improvements hold even on top of a strong baseline that uses a bagging strategy to reduce sensitivity to training data noise.

References

Berg-Kirkpatrick, Taylor, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in nlp. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and*

[8]The statistical significance tests used the points at the end of the P/R curves.

Computational Natural Language Learning, EMNLP-CoNLL '12, pages 995–1005, Stroudsburg, PA, USA. Association for Computational Linguistics.

Craven, Mark and Johan Kumlien. 1999. Constructing biological knowledge bases by extracting information from text sources. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology*, pages 77–86. AAAI Press.

Hoffmann, Raphael, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 541–550, Stroudsburg, PA, USA. Association for Computational Linguistics.

Min, Bonan, Xiang Li, Ralph Grishman, and Sun Ang. 2012. New york university 2012 system for kbp slot filling. In *Proceedings of the Fifth Text Analysis Conference (TAC 2012)*. National Institute of Standards and Technology (NIST).

Mintz, Mike, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 1003–1011, Stroudsburg, PA, USA. Association for Computational Linguistics.

Riedel, Sebastian, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD '10)*.

Sandhaus, Evan. 2008. The new york times annotated corpus. In *Linguistic Data Consortium, Philadelphia*.

Surdeanu, Mihai, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 455–465, Stroudsburg, PA, USA. Association for Computational Linguistics.

Diamonds in the Rough: Event Extraction from Imperfect Microblog Data

Ander Intxaurrondo†, Eneko Agirre†, Oier Lopez de Lacalle†, Mihai Surdeanu‡
†IXA NLP Group, University of the Basque Country
‡University of Arizona
{ander.intxaurrondo, e.agirre, oier.lopezdelacalle}@ehu.eus
msurdeanu@email.arizona.edu

Abstract

We introduce a distantly supervised event extraction approach that extracts complex event templates from microblogs. We show that this near real-time data source is more challenging than news because it contains information that is both approximate (e.g., with values that are close but different from the gold truth) and ambiguous (due to the brevity of the texts), impacting both the evaluation and extraction methods. For the former, we propose a novel, "soft", F1 metric that incorporates similarity between extracted fillers and the gold truth, giving partial credit to different but similar values. With respect to extraction methodology, we propose two extensions to the distant supervision paradigm: to address approximate information, we allow positive training examples to be generated from information that is similar but not identical to gold values; to address ambiguity, we aggregate contexts across tweets discussing the same event. We evaluate our contributions on the complex domain of earthquakes, with events with up to 20 arguments. Our results indicate that, despite their simplicity, our contributions yield a statistically-significant improvement of 33% (relative) over a strong distantly-supervised system. The dataset containing the knowledge base, relevant tweets and manual annotations is publicly available.

1 Introduction

Twitter is an excellent source of near real-time data on recent events, motivating the need for information extraction (IE) systems that operate on tweets rather than traditional news articles. However, using this data comes with its own challenges: tweets tend to use colloquial speech, noisy syntax and discourse, and, more importantly, the information reported is often inaccurate (e.g., reporting a different but similar magnitude for an earthquake) and ambiguous (e.g., reporting multiple potential earthquake locations, with insufficient context to guess which is the correct one).[1] The top rows in Table 1 show examples of these problems for an actual event in our dataset on earthquakes. This comes in contrast with "traditional" IE work on newswire documents, where information is considerably more accurate than microblog material, and none of the above observations hold (Grishman and Sundheim, 1996; Doddington et al., 2004).

As an example of the benefits of event extraction from a near real-time social-media resource, the last row in Table 1 lists a motivating example, where our system extracts the correct depth of an earthquake from the text tweeted by the U.S. Geological Survey, which is novel information that is missing in our manually-curated knowledge base.

In this work we take a classic event extraction (EE) task, where events are defined by templates containing a predefined set of arguments, and implement it using data from Twitter. We avoid the prohibitive cost of manual annotation through distant supervision (DS): we automatically generate train-

[1] We focus on microblogs here because they commonly contain inaccurate and/or ambiguous information. However, we believe that our contributions extend beyond microblogs because these innacuracies, especially inaccurate information, may appear in news article as well.

Approximate information	Earthquake in Honduras. So strong it strong it was felt in Guatemala as well. **7.1** offshore atlantic.
Ambiguous information	DTN **Indonesia**: *Peru* Earthquake Destroys Homes, Injures 100...
	6.9 magnitude earthquake rocks *Peru*.
	U.S.G.S. reports 6.9 Earthquake in *Peru*. NO TSUNAMI threat to Hawaii.
Information not in the knowledge base	#Earthquake M 7.0 – Ryukyu Islands, Japan T20:31:27 UTC , 25.95 128.40 depth: **22 km** <USGS URL> Local tsunami alert issued

Table 1: Challenges and opportunities for event extraction from Twitter. The first row shows a tweet with approximate information (in bold); the correct magnitude is 7.3 (cf. Table 2). The second row shows a first tweet with ambiguous information, which leads our baseline model to extract the incorrect country (in bold; correct country is *Peru*). The following two tweets help disambiguate the context. The last row shows a tweet containing information (in bold) that is missing in the knowledge base.

ing data by aligning a knowledge base of known event instances with tweets (Mintz et al., 2009; Hoffmann et al., 2011), which is then used to train a supervised extraction model (sequence tagger in our case). In seminal work on event extraction, (Benson et al., 2011) applied DS to both detect tweets about local events and then extracted values about two arguments (artist and venue). In our work, we work on automatically selected tweets, and scale the task to complex events with a large number of arguments. We focus on the domain of earthquakes, where each event has up to 20 arguments. Table 2 summarizes this task.

The contributions of this work are the following:

1. To our knowledge, this is one of the first works that analyzes the problem of distantly supervised extraction of complex events with many arguments from microblogs.

2. Our analysis shows (Section 3) that the biggest barrier is that information on Twitter can be *inaccurate* (containing approximately correct event argument values) and *ambiguous* (with insufficient context for accurate extraction). The top two blocks in Table 1 show an example of each. These challenges impact both evaluation and system development.

3. The analysis also highlights the need to adapt evaluation metrics to approximately correct information, which may appear both in text and in the knowledge base itself. For example, for a particular earthquake, the USGS reports a depth of 22 km., while NOAA reports 25 km[2]. We propose a new evaluation metric that gives partial credit to extracted argument values based on their similarity to existing values in the knowledge base.

4. We introduce two simple strategies that address the above barriers for system development: *approximate matching*, which addresses inaccurate values by allowing the distant supervision process to map values from the knowledge base to text even when they do not match exactly; and *feature aggregation*, which responds to small, ambiguous contexts by aggregating information across multiple tweets for the same event. For example, the first strategy considers the 7.1 magnitude in the first tweet in Table 1 as a training example because it is close to the value in the knowledge base (7.3). The second strategy classifies all instances of *Peru* jointly using a single set of features, extracted from all available tweets for the corresponding earthquake. For example, this feature set contains three values for the feature `previous-word` (*:*, *rocks*, and *in*). Each approach yields 19% relative improvement, 33% in combination.

5. We release a public dataset containing a knowledge base of earthquake instances and corresponding tweets for each earthquake[3].

2 Experimental framework

In this section we detail the creation of the knowledge base of earthquake events, the collection process for potentially-relevant tweets, and, lastly, our distant supervision framework, which serves as a platform for our contributions (Sections 5 and 6).

2.1 Knowledge base and tweet dataset creation

The **knowledge base (KB)** was created from the list of globally significant earthquakes during the 21st century, as reported by Wikipedia.[4] We se-

[2] `http://bit.ly/aq9Vxa` and `http://1.usa.gov/1p1gELB`

[3] `http://ixa.eus/Ixa/Argitalpenak/Artikuluak/1425465524/publikoak/earthquake-kb-dataset.zip`

[4] `https://en.wikipedia.org/wiki/List_of_21st-century_earthquakes`. Accessed on July 9th,

Argument Name	Arg. Type	# KB Values	Example Values	# DS Values	# MA Values
Date	D	108	2009-5-28	291	706
Time	T	108	T08:24:00	378	589
Country	L	108	Honduras	6294	6327
Region	L	77		2598	2663
City	L	77		1426	1723
Latitude	N	108	16.733	2	28
Longitude	N	108	-86.22	4	28
Dead	N	71	7	143	984
Injured	N	39		22	192
Missing	N	8		-	18
Magnitude	N	108	7.3	933	3403
Depth (km)	N	99	10	27	313
Countries affected(*)	L	37	Guatemala, Belize	436	357
Regions affected(*)	L	4		-	36
Landslides	B	8		7	9
Tsunami	B	10		408	273
Aftershocks	N	20		5	22
Foreshocks	N	3		6	-
Duration	T	7		-	1
Peak accel.	N	8		-	-
TOTAL		1,116		13,562	17,672

Table 2: Event arguments and types in the earthquake domain (first and second column), summary statistics for the knowledge base, i.e., the gold truth (third column), and values for one example earthquake (4th column). (*) indicates multi-valued arguments (all other are single-valued). The two rightmost columns give statistics for the number of mentions in the tweets per argument, as obtained through manual annotation (MA) or distant supervision (DS) (cf. Section 2.4). The argument types are the following: *D* date, *T* time, *L* location, *N* numeric, and *B* boolean.

lected earthquakes from the beginning of 2009, with the last reported earthquake happening on July 7th, 2013, and constructed the KB from the above Wikipedia list page and the individual infoboxes. Where necessary, argument values were normalized.[5] See Table 2 for a summary and an example.

We used the Topsy API[6] to search for **tweets** that are potentially relevant for each earthquake. We formed a query using the word "earthquake" plus the location, encoded as a disjunction of city, region, and country arguments. We retrieved tweets from the day before the date and time of the earthquake, up to seven days after. This procedure might also retrieve tweets about aftershocks, which we consider to be different events. We applied an aggressive method to discard aftershock tweets: we only kept tweets up to the first tweet that mentions a time expression more than a minute different from that of the main earthquake (after adjusting for time zone). For example, this heuristic removes all tweets starting with *"A 4.9 earthquake occurred in Ryukyu Islands, Japan on 2010-2-27 T10:33:21 at epicenter."* because the main earthquake occurred on February 26th at 8:31PM UTC. It is important to note that identifying event-relevant tweets is not the focus of this work (hence the simple heuristics used for tweet extraction). We focus instead on the *extraction* of information from such tweets. In a complete system, our approach would follow a component that detects event tweets automatically (Benson et al., 2011). The final dataset contains 108 earthquakes and 7,841 tweets, 72 tweets per earthquake on average, a maximum of 654 and a minimum of 2. 19 earthquakes had less than 10 tweets.

2.2 Manual annotation of tweets

In order to analyze the challenges faced by our EE system based on distant supervision, we also manually annotated all tweets.[7] The manual annotation included any mention of an event argument in the tweets. This included information already in the KB, but also information that is missing, caused by: variations of dates and times, similar but not identical latitude/longitude values, different reported numbers for dead/injured/missing etc. The first tweet in Table 1 is an example of this situation: even though the reported magnitude is different from the value in the KB (cf. example in Table 2), it was annotated during this process. In total, we annotated 17,672 mentions (at an average of two event arguments per tweet). Table 2 shows the breakdown per argument (the MA column), compared to the automatic annotations generated through distant supervision (the DS column). Note that some of the arguments have a very different coverage in the tweets compared with the KB. For example, latitude and longitude are rarely present in tweets, but affected countries are commonly mentioned. The quality of the manual annotation was assessed on a 5% sample of the dataset, which was annotated by an additional expert. The agreement was very high: 90% ITA and 85% Fleiss Kappa. Disagreements were generally

2013, at 2PM CET.

[5] Time and date expressions were converted to TimeML. Numerical values in English were converted to numbers, latitude and longitudes were converted to decimal format.

[6] `http://api.topsy.com/doc/`

[7] These manual annotations are used solely for post-hoc analysis, *not* to train our system.

due to missed argument mentions. Note that the cost of annotation was around 75 hours, confirming the cost-saving properties of distant supervision.

2.3 Dataset and experiment organization

We sorted the list of earthquakes in the KB chronologically, and chose the earliest 75% of the earthquakes as the training dataset, and the most recent (25%) for testing. The training set contained 81 earthquakes and their corresponding 6078 tweets, while the testing set contained 27 earthquakes and 1763 tweets. All development experiments were performed using 5-fold cross-validation over the training partition, where the folds were organized randomly by earthquake. Each fold contained tweets for around 15 earthquakes, but the number of tweets varied widely, with one fold having 585 tweets and another 2229.

The evaluation compares the argument values induced by our system with those in the gold KB, and computes precision, recall and F1 using the official scorer from the Knowledge Base Population (KBP) Slot Filling (SF) shared task (Surdeanu, 2013). We also incorporated the notion of equivalence classes proposed in the SF task. For instance, if the system predicted *Guerrero State* for the argument `region`, when the KB contains just *Guerrero*, we consider this result correct because the two strings are equivalent in this context. Our equivalence classes also include countries, regions, and cities with hashtags, unnormalized temporal expressions, etc. Where applicable, we checked statistical significance of performance differences using the bootstrap resampling technique proposed in (Berg-Kirkpatrick et al., 2012), in which we draw many simulated test sets by sampling with replacement from the set of earthquakes in the test partition.

2.4 Distant supervision for event extraction

For the initial extraction experiment, we followed a traditional distant supervision approach (Mintz et al., 2009), which has four steps: the KB of past events is aligned to the text; a supervised system is trained on the resulting annotated text; the system is run on test data; and the output slot values are inferred from the annotations produced by the system. We thus started by aligning the information in the KB to the training tweets using strict matching[8]. Table 2 compares the number of mentions automatically generated through DS against the number of manually annotated mentions. As expected, the strict matching criterion yields fewer mentions than the manual annotation.

As an example of this process, given the Honduras earthquake in Table 2, this procedure will annotate two argument mentions in the first tweet from Table 1, `country` and `affected-country`, as follows:

> Earthquake in <country>Honduras</country>. So strong it was felt in <affected-country>Guatemala</affected-country> as well. 7.1 offshore atlantic.

Note that the magnitude in the tweet is different from the one reported in the KB and it will thus be left unmarked (we revisit this issue in Section 5).

Using this automatically-generated data, we trained a sequential tagger based on Conditional Random Fields (CRF)[9]. Based on the output of the CRF, we inferred the arguments values using noisy-or (Surdeanu et al., 2012), which selects the value with the largest probability for each single-valued argument by aggregating the individual mention probabilities produced by the CRF.[10] In the case of multi-valued arguments (`affected-country` and `affected-region`) we choose all values that had been annotated by the sequential tagger.

3 Initial results and analysis

The left block in Table 3 reports the results on development (5-fold cross-validation) of the initial event

[8] We identified two types of arguments: those that have binary (yes/no) values (`tsunami` and `landslides`) and those having other values. For the first type, we search the tweets corresponding to the target earthquake for a small number of strings (e.g., *tsunami* and *tsunamis*), and annotate all matches (e.g., <*tsunami*> *tsunami* </*tsunami*>). For non-binary valued arguments, we searched the tweets for exact occurrences of the corresponding values, and annotated all matching strings. When the same value appears in more than one argument for the same earthquake (e.g., *7* as both magnitude and number of dead people), we choose the most common label (e.g., magnitude cf. Table 2).

[9] We used the linear CRF in Stanford's CoreNLP package, with the default features (word form, PoS, lemma, NERC) for the `macro` configuration: `http://nlp.stanford.edu/software/corenlp.shtml`.

[10] For multi-token mentions (e.g. *New Zealand*) we use the average of the token probabilities.

	Strict Evaluation			Lenient Evaluation		
System	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
DS-CRF	53.1	22.0	31.1	67.4	27.9	39.4
MA-CRF	44.1	26.1	32.8	62.1	36.8	46.2

Table 3: Development: Results for the distant supervision system (DS-CRF). We also include results for the same CRF trained on manual annotations (MA-CRF). The regular evaluation is shown in the left columns and lenient evaluation (cf. Section 4) in the right.

extraction system based on a distantly-supervised CRF (DS-CRF), which notably attains higher precision than recall. These results are fair, e.g., they are comparable to those of (Benson et al., 2011), even though their events had much fewer argument types than ours (two vs. twenty). More importantly, we use this system's output to analyze where the approach could be improved. For the sake of comparison, we trained the same CRF with the manually annotated tweets, cf. Section 2 (MA-CRF). The MA-CRF results in Table 3 indicate that the main loss when doing distant supervision is in recall, but the overall F1 is close. This is remarkable, as the much more expensive MA-CRF (75 hours of human annotation) is taken to be an upperbound for DS-CRF.

Manual inspection showed that that DS-CRF returns fewer argument values than MA-CRF (328 vs. 469), from "easier" (more common) arguments which have a higher chance of appearing both in the text and the KB. Importantly, MA-CRF has lower precision than its distant supervision counterpart because it is trained on manual annotations, which included many mentions not in the KB. The consequence of this strategy is that MA-CRF tends to produce spurious mentions (i.e., mentions not in the KB) at evaluation time, which lowers precision.

In addition, we analyzed the annotations created through distant supervision[11], which produced 13,562 argument mentions in the training tweets (cf. Table 2, which also includes a breakdown by argument). This data contains incorrectly annotated strings (false positives) and also misses relevant argument values (false negatives). A comparison of these DS annotations against the manual annotations on all training tweets (17,672 mentions) yielded that 97.4% were correct, but that 27.4% of the gold manual annotations were missed. This is an important result: it demonstrates that, unlike in the problem of relation extraction (RE) where the major issue is the large percentage (higher than 30%) of false positives in automatically-created annotations (Riedel et al., 2010), here the fundamental roadblock is missing annotations (i.e., false negatives). We explain this difference by the fact that for this event extraction domain, it is trivial to identify domain-relevant tweets, which reduces the number of false positives for event arguments. We believe this generalizes to many other EE domains, e.g., airplane crashes (Reschke et al., 2014) or terrorist attacks, where the event context can be summarized accurately with a small number of keywords (e.g., flight number and date for the airplane crashes domain).

We also did a post-hoc analysis of the quality of the arguments induced by DS-CRF. One of the most significant outcomes of the analysis is that a large portion of numeric values (31.3%) were partially correct, in that the returned values were very similar to those in the KB (see for instance the 7.1 vs. 7.3 example in Section 1). This strongly suggests that the evaluation metric should be more lenient, and give credit to argument values that are similar to the gold ones.

4 Lenient evaluation

The previous analysis suggests that traditional evaluation measures unnecessarily penalize arguments containing values that do not match the gold truth exactly. Rather than giving no credit when predicted values are different from gold ones, we devised a simple extension to the KBP evaluation measures that take into account the similarity between the values of system and gold arguments, where the similarity depends on the type of each slot (cf. Table 2). For numeric values, we use the following formula, where x is the predicted value, and g the gold value:

$$sim(x, g) = max\left(1 - \frac{|x-g|}{g}, 0\right)$$

For example, given a gold value of 7.3, a system value of 7.2 would have a similarity of 0.98, and a system value of 14.6 or larger would have a similarity 0. If both values are equal, similarity is 1.

For the other slot types, the similarity function is

[11] Note that these are the argument mention annotations used to train DS-CRF, not the arguments inferred by the DS-CRF system.

discrete, with values set to 1 (proposed slot is correct) or 0 (incorrect) as follows. We consider a proposed *temporal* argument as correct if it is within a span of 5 minutes of the corresponding gold temporal value. *Durations* are judged as correct if they are within 10 seconds of the gold values. We considered proposed *dates* as correct if they differ by at most one day from the gold date.[12]

For *location* arguments, we use GeoNames[13] to obtain the coordinates of the locations produced by the system that do not match the information in the KB. Based on the average size of countries, regions, and cities, we consider these additional locations as correct if they are at the following distance (or closer) from the gold locations: 500 kms for countries, 50 kms for regions, and 10 kms for cities.

The original KBP scorer increases the value of True Positives (TP) by 1 every time a predicted argument matches its gold value. In the proposed lenient scorer, TP is increased by the similarity between the predicted and gold values. The precision and recall will be thus calculated as follows (*SYS* for number of predicted argument values, *GOLD* for number of gold argument values):

$$prec = \frac{\sum sim(x,g)}{SYS}, \; rec = \frac{\sum sim(x,g)}{GOLD}$$

The right block in Table 3 lists the results under this lenient evaluation for the experiment initially reported in the left block in the same table. As expected, these results are higher than the ones using the strict measure, but maintain the relative order of the systems in each of the evaluation measures. The difference in precision between DS-CRF and MA-CRF decreases, indicating that the new measure assigns partial credit to the larger amount of argument values extracted by MA-CRF. The difference in recall values remains large. We address this in the next section.

5 Approximate distant supervision

The previous section demonstrated that many tweets contain argument values which are similar but not identical to the data in the knowledge base. These values would not be annotated during alignment by traditional distant supervision, which expects an exact match between knowledge base values and tweet texts. This means that DS-CRF will be trained with less data than what is available (e.g., without the 7.1 magnitude example in the tweet in Section 2.4). Here we demonstrate that a simple extension to distant supervision that annotates values close to the values in the knowledge base, results in improved performance.

The proposed alignment algorithm scans the training tweets, and labels named and numeric entities as positive argument examples (with the corresponding label from the KB), if they are deemed similar to the gold values according to the similarity formulas introduced in the previous section. This is a trivial process for discrete similarities, but requires some care for continuous similarity functions, which are triggered for numeric arguments. In this situation, numeric entities are considered as positive examples only if their similarity function returns a value over a certain threshold with a known argument in the KB. If a numeric mention has more than one matching argument in the KB, the algorithm chooses the argument label with the highest simi-

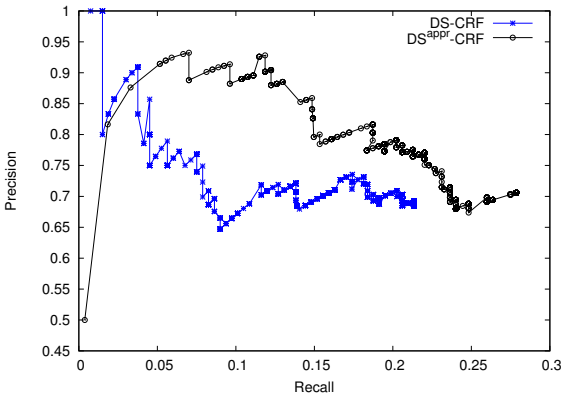

Figure 1: Test: Precision/Recall curves for regular DS and approximate DS on test (lenient evaluation).

System	Prec.	Rec.	F1
DS-CRF	68.4	21.3	32.5
DS^{appr}-CRF	70.6	27.8	39.9 †

Table 4: Test: Regular (DS-CRF) and approximate DS (DS^{appr}-CRF) results, with lenient evaluation. † indicates statistically significant improvement over DS-CRF ($p < 0.05$).

[12]These thresholds might change in other domains, but adjusting these values is trivial.

[13]http://www.geonames.org/

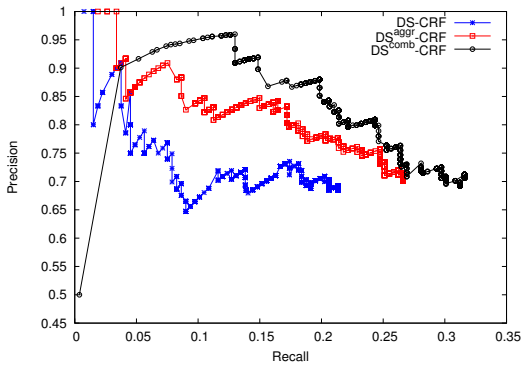

Figure 2: Test: P/R curves for DS-CRF, feature aggregation and combination with approximate DS (lenient evaluation).

System	Prec.	Rec.	F1
DS-CRF	68.4	21.3	32.5
DS^{aggr}-CRF	70.1	26.6	38.6 †
DS^{comb}-CRF	69.2	31.2	43.1 †
MA-CRF	69.1	37.9	48.9

Table 5: Test: Results for regular DS (DS-CRF), DS with feature aggregation (DS^{aggr}-CRF), and the DS model that combines feature aggregation and approximate matching (DS^{comb}-CRF), with lenient evaluation. † indicates statistically significant improvement over DS-CRF ($p < 0.05$). We include the results of the CRF trained on manual annotations (MA-CRF) as a performance ceiling for this task.

larity value; if all have the same similarity, the algorithm chooses the most frequent label in training.

We tuned the threshold hyper parameter for numeric values over the training dataset using 5-fold cross validation, which yielded 0.95 as the optimal value. Table 4 shows the results for the test partition using this threshold, and Figure 1 shows the corresponding P/R curves. Both results are generated using the proposed lenient evaluation. The results in the table show that, despite its simplicity, the proposed alignment algorithm yields considerable, statistically-significant improvements. The P/R curves show that the improvement holds for all recall points[14].

[14]The curves for the strict evaluation are similar, and were omitted for brevity.

6 Feature aggregation

The second block in Table 1 illustrates a common scenario on Twitter, where a short, ambiguous tweet derails the extraction. We address this problem of insufficient local context with a method inspired by work in relation extraction, where relation instances between identical entities are classified jointly using the conjunction of features from all instances (Mintz et al., 2009). We adapt this idea to our sequence tagging EE model as follows:

1: We focus on location, date and temporal entities (both earthquake time and duration) which are argument candidates that are often ambiguous, i.e., they may be classified as more than one argument type. For example, a location entity may be labeled as `country`, `region`, `country-affected`, etc. We exclude numeric entities due to potential feature collisions between different argument types: we observed that, in training, several earthquakes had different numeric arguments with the same value. For example, the magnitude and depth of the 2012 Zohan earthquake were 5.6. Applying feature aggregation to examples of these arguments would lead to collisions between features from different classes.[15]

2: For each token that appears in one of these named entities, we identify all its instances across the relevant tweets, and share features across all these token instances. For example, for the tweets in the second block in Table 1, our approach identifies *Peru* as an argument mention candidate. All three instances of *Peru* are then classified using the same shared features, e.g., using three values for the feature `previous-word` (:, *rocks*, and *in*). This process is repeated for each earthquake individually, because tokens may be labeled differently in different earthquakes. This approach produced 37% more features than the DS-CRF baseline.[16]

The positive effect of feature aggregation is confirmed by the formal evaluation on the test dataset.

[15]Initial experiments confirmed this hypothesis: feature aggregation did not improve results for numeric arguments in development. In future work, we will explore multi-instance multi-label algorithms to handle this situation (Surdeanu et al., 2012).

[16]We also tried skip-chain CRFs (Getoor and Taskar, 2007), but found that our simpler approach converges considerably faster and produces slightly better results. We do not show those results for brevity.

Table 5 shows a statistically significant improvement in overall F1, for the lenient evaluation. The P/R curves (Fig. 2) indicate that DS^{aggr}-CRF's improvement comes from both better recall and better precision that the DS-CRF baseline.

Table 5 and Fig. 2 also show that the combination of approximate matching and aggregation outperforms the individual models, demonstrating that feature aggregation is complementary to approximate matching. The combined model attains a relative improvement of 33% over the DS-CRF baseline, reaching approximately 88% of the ceiling performance for this task (MA-CRF row, the CRF trained on manual annotations).

7 Related work

There has been considerable recent interest in IE from Twitter. However, in general, these works use supervised learning frameworks (Popescu et al., 2011; Ritter et al., 2012), and/or they use either a coarse representation of events, which reduces to topic modeling or classification of entire tweets (Popescu et al., 2011; Becker et al., 2011; Ritter et al., 2012), or a simplified representation of events with few arguments (Sakaki et al., 2010; Popescu et al., 2011; Benson et al., 2011; Ritter et al., 2012). In contrast, our work uses a complex event representation with 20 arguments, and does not require any manual annotation of tweets. Our work is closest, but complementary to the work of (Benson et al., 2011), which also uses distant supervision for event extraction: We provide solutions for two problems they do not address (inaccurate and ambiguous information) and we focus on more complex events (20 arguments vs. two).

This paper is also complementary to systems which detect event-relevant tweets (Sakaki et al., 2010; Petrović et al., 2010). In future work, we plan to replace our simple method of extracting relevant tweets by one of these approaches, producing a system that monitors microblogs in realtime to automatically construct event-specific knowledge bases.

Our work uses the framework of distant supervision, which has also received considerable attention recently. Nevertheless, most of these works focus on the extraction of binary relations from well-formed documents (Mintz et al., 2009; Riedel et al., 2010; Hoffmann et al., 2011; Surdeanu et al., 2012). We use the much noisier Twitter as the underlying text, and extract complex events instead of binary relations. We note, however, that the idea of feature aggregation is inspired by these works (Mintz et al., 2009; Riedel et al., 2010), but, to our knowledge, we are the first to apply it to event extraction and sequence tagging. In the DS space, our work is closest to (Reschke et al., 2014), which use it to extract complex events (airplane crashes) from newswire text. Because they focus on newswire, they do not need to address the potential for inaccurate or ambiguous information, which is the main focus of our work.

System	Prec.	Rec.	F1
DS-CRF	66.21	20.66	31.49
DS^{aggr}-CRF	68.27	25.92	37.58 †
DS^{comb}-CRF	61.53	27.61	38.25 †
MA-CRF	68.76	27.61	39.40

Table 6: Test: Replica of the experiments in Table 5 using a threshold of 0.95 for the lenient evaluation measure. All other settings are identical to the experiments in Table 5. † indicates statistically significant improvement over DS-CRF ($p < 0.05$).

8 Discussion: An alternate evaluation measure

Designing relevant measures for lenient evaluations, such as the one discussed here, is an open research issue. For example, the method proposed in Section 4 gives partial credit to all reported (positive) numeric values in the interval $[0, 2g]$, where g is the correct value for the corresponding slot (see the equation in Section 4). But other, stricter, measures are certainly possible.[17] For example, one stricter variant of our proposed measure would assign partial credit only for predicted values that have a similarity of 0.95 or higher with the gold truth (inline with our approximate DS training process). For example, for the same gold numeric value g, the measure assigns partial credit only for predicted values in the interval $[0.95g, 1.05g]$.

We repeated the experiments in Table 5 using this alternate evaluation measure. The result are summarized in Table 6. The results reported in Table 5 do

[17] We thank the anonymous reviewer for the suggestion.

not alter the findings of the paper. In fact, under this stricter evaluation measure, our results are stronger: DS^{comb}-CRF, which combines both our ideas, approaches with nearly 1 F1 point MA-CRF, which trains on manually annotated data.

9 Conclusions

To our knowledge, this is one of the first works that analyzes the problem of distantly supervised complex event extraction on microblogs. This near real-time data source is challenging, with inaccurate information and short, ambiguous texts, as shown by our empirical analysis of the dataset. We proposed two simple techniques to address these problems: (a) a novel distant supervision paradigm, which implements an alignment algorithm that allows text snippets that are similar but not identical to argument values in the knowledge base to be annotated (thus producing better training data); and (b) a feature aggregation strategy that provides richer information across tweets to cope with ambiguity. Our results on earthquake-related tweets show that each improvement yields 19% significant improvement when applied on top of a strong system based on sequence tagging (CRFs). We show that these contributions are complementary: a model that combines both performs better than each of the above individual models, with an improvement of 33% over the baseline. All in all, our approach attains approximately 88% of the ceiling performance for this task, which is obtained by a system trained on manually-annotated tweets, validating the hypothesis that distant supervision is useful for a complex event extraction task.

In addition, we devised a lenient evaluation measure which incorporates the similarity between the extracted argument values and the gold truth, rather than considering as correct only the extractions that exactly match the gold values. We show that this evaluation models the event extraction task better, and, furthermore, is more realistic, especially in view of imperfect knowledge bases.

Lastly, we release a dataset containing an event knowledge base constructed from Wikipedia information on earthquakes, which contains 108 earthquakes, 20 different argument types, and 1,116 argument values. The dataset also includes a collection of relevant tweets about these earthquakes, totaling 7,841 tweets. The dataset is publicly available.

References

Hila Becker, Mor Naaman, and Luis Gravano. 2011. Beyond trending topics: Real-world event identication on twitter. In *Proceedings of the Conference of the Association for the Advancement of Articial Intelligence*.

Edward Benson, Aria Haghighi, and Regina Barzilay. 2011. Event discovery in social media feeds. In *Proceedings of the Conference of the Association for Computational Linguistics (ACL)*.

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in nlp. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 995–1005, Stroudsburg, PA, USA. Association for Computational Linguistics.

G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. 2004. The automatic content extraction (ace) program – tasks, data, and evaluation. In *Proceedings of LREC*.

L. Getoor and B. Taskar, 2007. *Introduction to statistical relational learning*. MIT Press.

R. Grishman and B. Sundheim. 1996. Message understanding conference - 6: A brief history. In *Proceedings of the International Conference on Computational Linguistics*.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics*.

Saša Petrović, Miles Osborne, and Victor Lavrenko. 2010. Streaming first story detection with application to twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 181–189, Los Angeles, California, June. Association for Computational Linguistics.

Ana-Maria Popescu, Marco Pennacchiotti, and Deepa Paranjpe. 2011. Extracting events and event descriptions from twitter. In *Proceedings of the 20th International Conference on World Wide Web*.

Kevin Reschke, Martin Jankowiak, Mihai Surdeanu, Christopher D. Manning, and Daniel Jurafsky. 2014. Event extraction using distant supervision. In *Proceedings of LREC*.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD '10)*.

Alan Ritter, Mausam, Oren Etzioni, and Sam Clark. 2012. Open domain event extraction from twitter. In *Proceedings of KDD*.

Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 851–860, New York, NY, USA. ACM.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing and Natural Language Learning (EMNLP-CoNLL)*.

Mihai Surdeanu. 2013. Overview of the tac2013 knowledge base population evaluation: English slot filling and temporal slot filling. In *Proceedings of the TAC-KBP 2013 Workshop*.

Lurrikarei buruzko informazioa eskuratzen Twitter bidez.

Ander Intxaurrondo, Eneko Agirre eta **Oier Lopez de Lacalle**

Ixa Taldea. Euskal Herriko Unibertsitatea.

Laburpena

Lan honetan, mikroblogetatik gertaera konplexuak erauzten dituen sistema bat aurkezten dugu, urruneko gainbegiraketa erabiliz. Denbora errealeko datu-iturri hauetako testuak laburrak, sintaxi zaratatsukoak eta anbiguoak dira; baina informazio kantitate handiak topatu ditzakegu. Gure ekarpena lurrikaren domeinuan ebaluatzen dugu, 20 argumentutik gorako gertaerekin. Ezagutza-basea eta txio garrantzitsuak dituen datu-multzoa publikoki dago eskuragarri, biak ingelesez daude.

Hitz gakoak: Hizkuntzaren prozesamentua, gertaeren erauzketa, urruneko gainbegiraketa, ezagutza-baseak

Abstract

In this work, we introduce an event extraction approach that extracts complex event templates from microblogs, using distant supervision. These near real-time data source texts are short, ambiguous and contain dirty syntax; but we can find lots of information. We evaluate our contribution on the domain of earthquakes, with events with up to 20 arguments. The dataset containing the knowledge-base and relevant tweets is publicly available, both in English.

Keywords: Language processing, event extraction, distant supervision, knowledge-bases

1 Sarrera eta motibazioa

Twitter baliabide ona bilakatu da denbora errealean gertaera desberdinei buruko datuak lortzeko era azkarrean, informazio erauzketa (IE) ohiko egunkari-artikuluak ez diren beste informazio-iturrietan aplikatzera motibatuz. Era askotako informazioa eskuratu dezakegu, hala nola artista baten emanaldi bati buruzkoa, hegazkin istripuak, eta abar. Txioek hizkera kolokiala, sintaxi eta diskurtso zaratatsua, eta informazio anbiguoa izateko joera dute; hala ere, informazio kantitate handiak aurki ditzakegu.

Lan honetan gertaera erauzketa (GE) sistema bat garatu dugu. GE sistemak, testuetako gertaerak identifikatzen saiatzen dira, eta testuinguruko elementu desberdinek jokatzen duten rolak identifikatzen saiatzen dira. Aukeratu dugun domeinua lurrikarena da, eta lurrikara bakoitzeko 20 argumentu desberdini buruzko informazioa eskuratzen dugu, automatikoki aukeratutako txio sorta batekin.

Informazio erauzketa sistema onenetako corpusak eskuz etiketatzen dira, oso emaitza onak ematen dituzte, baina etiketatze-prozesu honen kostua oso garestia da. Lan honetan, eskuzko etiketazioaren kostu garestia alde batera uzten dugu, eta entrenamenduko corpusak automatikoki eskuratzeko algoritmo bat erabili: urruneko gainbegiraketa (UG).

Lan honetarako egindako ekarpenak ondorengoak dira:

1. Hau da urruneko gainbegiraketaren bidez mikroblogetatik argumentu askotako gertaera konplexuak erauzten dituen lehen lana.
2. Lurrikarei buruzko ezagutza-base bat jarri dugu publikoki eskuragarri, baita lurrikara bakoitzari dagozkion txioak ere. Txioak eta ezagutza-basea ingelesez daude.

Hasteko, lan honetan ezagutza-baseak eta urruneko gainbegiraketa zer diren azalduko dugu. Jarraian lurrikarei buruzko ezagutza-basea nola sortu dugun azaldu, eta lurrikara bakoitzari buruzko txioak nola eskuratu ditugun komentatuko dugu. Ondoren esperimentuak eta emaitzak erakutsiko ditugu. Amaitzeko, lan honi buruzko ondorioak eta etorkizuneko lanak aurkeztuko ditugu.

1 Irudia: Bi infotaulen adibideak.

(a) Bernardo Atxagaren infotaula.

Datu pertsonalak	
Izen osoa	Jose Irazu Garmendia
Ezizena	*Bernardo Atxaga*
Jaio	1951ko uztailaren 27a Asteasu, Gipuzkoa (Euskal Herria)
Bikotekidea(k)	Asun Garikano
Webgunea	http://www.atxaga.org/

(b) Lurrikara baten infotaula.

Date	15:40 PDT, April 4, 2010
Duration	89 seconds
Magnitude	7.2 M_w
Depth	10 kilometers (6 mi)
Epicenter	32.128°N 115.303°W
Countries or regions	Mexico United States
Max. intensity	IX[1]
Tsunami	No
Landslides	Yes
Aftershocks	Yes
Casualties	4 killed, at least 100 injured in the vicinity of Mexicali.[2]

2 Arloko egoera eta ikerketaren helburuak

Ezagutza-base bat (EB) ezagutza kudeatzeko datu-base berezi bat da. Ezagutzaren bilketa, antolaketa eta berreskurapena konputazionalki egiteko baliabideak hornitzen ditu. Azken urteetan informazio erauzketan eta lengoaia naturalaren prozesamenduan geroz eta gehiago erabiltzen dira. Gehien erabiltzen direnak DBpedia[1] eta Freebase[2] dira.

Ezagutza-baseak hainbat kontzeptu eta entitateren multzoak dira, eta entitate hauei buruzko informazioa era eskematikoan eta ulergarrian irudikatzen dute. Entitate bakoitzak erlazio batzuk ditu, erlazio bakoitzak izen bat jasotzen du eta beste entitate, kontzeptu edo balio batekin erlazionatuta dago. Ezagutza-baseetan aurki ditzakegun entitateak pertsonak, erakundeak, lekuak, denbora-adierazpenak eta beste hainbat motatakoak izan daitezke, bai entitate nagusia, baita erlazioan parte hartzen duen bigarren entitatea ere.

Wikipediako infotaulak oso baliagarriak dira ezagutza-baseak sortzeko. Infotaulak Wikipediako artikulu baten eskubialdean aurki ditzakegu, artikuluko informazioaren laburpen bat emanez. 1a irudian Wikipediako Bernardo Atxaga idazlearen artikuluko[3] infotaula dugu, bertatik ondorengo erlazioak eskuratu ditzakegu, besteak beste:

- Bernardo Atxaga - *jaioteguna* - 1951ko uztailaren 27a
- Bernardo Atxaga - *jaioterria* - Asteasu
- Bernardo Atxaga - *izen_osoa* - Jose Irazu Garmendia

Urruneko gainbegiraketa (UG) (Mintz *et al.*, 2009) lanean erlazio erauzketarako proposatutako paradigma bat da. Hurbilketa honek automatikoki etiketatzen ditu corpusak. UGren motibazio nagusia eskuzko lanak sahiestea da, hala nola corpusen eskuzko etiketatzea.

UGren arabera, ezagutza-base batek bi elementuren artean erlazio bat dagoela zehazten badu, eta bi elementu hauek esaldi berean agertzen badira, esaldi horrek erlazio hori adieraziko du nola edo hala.

Corpus batetik Bernardo Atxagari buruzko esaldiak berreskuratu ondoren, esaldi hauetan dauden entitate desberdinak detektatu behar ditugu. 1 taulan Bernardo Atxagari buruzko esaldi desberdinak ditugu, aurretik aipatutako erlazioak adieraziz.

Urruneko gainbegiraketa ia ez da erabili gertaerei buruzko informazioa erauzteko. (Benson *et al.*, 2011) da GE eta UG batu dituen lehen lana, Twitterreko txioak erabiliz. Esperimentu hauetan, astista des-

[1] `http://dbpedia.org/About` . Euskaraz `http://eu.dbpedia.org/index.php?title=Azala`
[2] `https://www.freebase.com/`
[3] `http://eu.wikipedia.org/wiki/Bernardo_Atxaga`

1 Taula: Bernardo Atxagari buruzko esaldi desberdinak, eskuineko zutabeak idazlea eta letra lodiz jarritako elementuen arteko erlazioa adierazten du.

Esaldia	**Erlazioa**
Bernardo Atxaga 1951ko uztailaren 27an jaio zen Asteasu herrian.	*jaioteguna*
Bernardo Atxaga, **1951eko uztailaren 27an** jaioa, idazle ospetsu bat da.	*jaioteguna*
Bernardo Atxaga 1951ko uztailaren 27an jaio zen **Asteasu** herrian.	*jaioterria*
Asteasu da **Bernardo Atxaga** jaio zen herria.	*jaioterria*
Bernardo Atxaga izengoitiz, agiri ofizialetako izen-deiturez **Jose Irazu Garmendia** (...)	*izen_osoa*
Bernardo Atxaga da **Jose Irazu Garmendiaren** goitizena.	*izen_osoa*

berdinek New York hirian egindako emanaldiei buruzko informazioa lortzen saiatzen dira, baina bakarrik emanaldi bat non egin duten jakin nahi dute, emanaldiari buruzko informazio sakonagoa (ordua, ikusle kopurua,...) alde batera utzita. Gure gertaera erauzketa esperimentuetan berriz, lurrikarei buruzko informazio asko erauzten dugu.

(Reschke *et al.*, 2014) lanean ere UG erabiltzen dute gertaerak erauzteko. Lan honetan, hegazkin istripuei buruzko hainbat informazio erauzten dute: eguna, istripu-mota, istripua gertatu den lekua, hegaldi-zenbakia, hildakoak eta abar. Gure lana eta hau oso parekoak dira, baina beraiek berri agentzien dokumentuak aztertuz eskuratzen dute informazio hori, guk ordea, Twitter erabiltzen dugu.

UG gertaera erauzketan aplikatzeko, aurretik testuetako gertaerak identifikatzea komeni da. UGren algoritmoa ezin dugu zuzenean aplikatu GErako, gertaeraren izena ez delako esaldietan esplizituki aipatzen. UG gertaeren erauzketara moldatzeko, ondorengo heuristikoa proposatzen dugu: esaldi bat gertaera konkretu bati buruzkoa bada, esaldian dagoen aipamen batek batek ezagutza-baseko argumentu baten balio berdina badu, aipamen horrek argumentu mota hori adieraziko du nola edo hala.

3 Ikerketaren muina

Atal honetan, lurrikarei buruzko EBa nola sortu dugun azalduko dugu, txioak eskuratzeko jarraitutako irizpideekin batera. Sistemak txio bakoitza nola prozesatu duen azalduko dugu, eta amaitzeko emaitzak erakutsi.

3.1 Lurrikarei buruzko ezagutza-basearen sorkuntza

Lan honetarako, ingelesezko Wikipediako infotauletan oinarritutako ezagutza-base bat sortu dugu. Ezagutza-base honetan 2009ko hasiera eta 2013ko uztailaren arteko lurrikarak aurki ditzakegu. EB honetan, lurrikarari buruzko hainbat informazio dago bilduta, hala nola eguna, lekua, magnitudea eta abar. Guztira 108 lurrikara desberdinei buruzko informazioa bildu dugu.

1b irudian, ingeleseko Wikipediako infotaula bat dugu. Infotaula hau Mexikoko Baja Californian[4] gertatutako lurrikara batena da.

Ezagutza-basea 20 argumentu desberdinez osatuta dago. 2 taulak argumentu horiek biltzen ditu, argumentu bakoitzaren datu-mota zein den adieraziz, honen esanahia euskaraz, eta aurretik adibide bezala erabili dugun lurrikararen datuekin. Argumentu-motak ondorengoak dira: *E* eguna, *D* denbora, *L* lekua, *Z* zenbakizkoa eta *B* boolearra (bai ala ez). Asteriskoa (*) duten argumentuek balio bat baino gehiago onartzen dute. 4. zutabeak argumentu bakoitzeko zenbat informazio dugun ezagutza-basean adierazten du; ikus dezakegunez, lurrikara guztiek dute eguna, ordua, estatua, magnitudea eta koordenatu geografikoei buruzko informazioa; aurrelurrikarak, iraupena, desagertu-kopurua eta beste argumentu batzuei buruzko informazioa 10 lurrikara baino gutxiagotan aurki dezakegu. Azken zutabea hurrengo azpiatalean azalduko dugu.

[4] `http://en.wikipedia.org/wiki/2010_Baja_California_earthquake`

2 Taula: Lurrikarei buruzko ezagutza-basearen argumentuak, hauen esanahia euskaraz, argumentu-motak, adibide bat, argumentu bakoitzaren balio-kopurua EBan, eta urruneko gainbegiraketaren bidez zenbat aldiz etiketatu dugun argumentu bakoitza datu-multzoan.

Argumentua	**Euskaraz**	**Mota**	**Adibidea**	**# EB**	**# UG**
date	Eguna	E	2010-4-4	108	291
time	Ordua	D	T22:40:00	108	378
country	Estatua	L	Mexico	108	6294
region	Herrialdea	L	Baja California	77	2598
city	Hiria	L	-	77	1426
latitude	Latitudea	Z	32.128	108	2
longitude	Longitudea	Z	-115.303	108	4
dead	Hilkakoak	Z	4	71	143
injured	Zaurituak	Z	100	39	22
missing	Desagertuak	Z	-	8	-
magnitude	Magnitudea	Z	7.2	108	933
depth (km)	Sakonera (km)	Z	10	99	27
affected-country(*)	Estatu kaltetua	L	United States	37	436
affected-region(*)	Herrialde kaltetua	L	-	4	-
landslides	Lubiziak	B	yes	8	7
tsunami	Tsunamiak	B	-	10	408
aftershocks	Erreplikak	Z	-	20	5
foreshocks	Aurrelurrikarak	Z	-	3	6
duration	Iraupena	D	00:01:29	7	-
peak-acceleration	Azelerazio sismikoa	Z	-	8	-
			Guztira	**1116**	**13562**

3.2 Txioak Twitterretik eskuratzen

Lurrikarei buruzko txioak eskuratzeko, Topsy Labs[5] enpresaren baliabideak[6] erabili ditugu. Enpresa hau baliabide sozialen edukien bilaketa eta analisira jarduten da.

Lurrikara bakoitzeko bilaketak egitean, *earthquake* hitz-gakoa erabili dugu, ezagutza-basean agertzen zen kokapenarekin (hiriak, herrialdeak eta estatuak) batera zehaztuz. Lurrikara gertatu baino egun bat lehenago eta hortik 7 egun geroago idatzitako txioak eskuratzen ditugu bakarrik.

Lurrikara gertatu baino egun bat lehenagoko tweetak eskuratzeak arrazoi bat dauka: ordu-eremuak[7]. Txioak ez daude geolokalizatuta, kontuan hartu behar da txiolariak ez direla profesionalak eta txiokatzean lurrikara gertatutako unea aipatzeko beren bizilekuko ordua erabiltzen dutela denbora estandarraren ordez.

Lortutako txio asko lurrikaren erreplikei buruzkoak dira. Erreplikak lurrikara nagusiaren ondoren gertatutako beste lurrikara batzuk dira, hauek epizentrotik gertu daude eta normalean nagusiak baino magnitude txikiagoa dute. Erreplikak direla eta, txioetako informazioan eta EBko informazioan kontraesanak egongo dira, sistemaren ikasketa prozesua nahastuz eta ebaluatzean emaitza okerrak itzuliz.

3.2.1 Erreplikak antzematen

Erreplikak gertaera desberdin bezala tratatu ditugu, eta ezagutza-basean zeuden lurrikarei buruzko txioak bakarrik edukitzearren, metodo oldarkor bat aplikatu dugu erreplikei buruzko txioak baztertzeko. Heuristiko hau aplikatzeko, txioak kronologikoki ordenatu ditugu. Heuristiko hau txio desberdinetan aipatzen diren denbora-adierazpenetan oinarritzen da:

[5] `http://topsy.com`
[6] `http://api.topsy.com/doc`
[7] `http://eu.wikipedia.org/wiki/Ordu-eremu`

1. Txioetan lurrikara bakoitzeko lortutako lehen denbora-adierazpena gordetzen dugu. *ordua:minutua* patroia erabili da denbora-adierazpen hauek antzemateko. Segunduak ez ditugu kontuan hartzen.
2. Geroagoko txio batean agertzen den denbora-adierazpena lehenengoarekiko desberdina bada, bai ordua bai minutua, txio hau erreplika bati buruz ari dela ulertzen dugu. Txio hau eta ondoren datozen beste guztiak kentzen ditugu, hemendik aurrera jasoko ditugun txioak erreplika horri edo beste batzuei buruzkoak izango direlakoan. Ordua lurrikara nagusiko orduarekiko desberdina bada baina minutua berdina, orduan lurrikara nagusitzat hartzen dugu txio hau, denbora eremu desberdin batean dagoen pertsona batek txiokatu duelakoan.

Bukaerako datu-multzoak 108 lurrikara desberdin ditu eta guztira 7841 txio desberdin. Batazbesteko 72 txio ditugu lurrikara bakoitzeko, gehienez 654 txio eta gutxienez 2 edukiz. 19 lurrikarek 10 txio baino gutxiago dituzte.

3.3 Aipamenen etiketatzea txioetan

Urruneko gainbegiraketaren algoritmoa aplikatuz, lurrikara bakoitzaren txioak bildu eta EBko argumenturen baten balioarekin bat egiten duten aipamenak etiketatu ditugu. Adibide bezala, Baja Californiako lurrikarari buruzko txio bat hartuko dugu:

- Update : Earthquake in Baja California, Mexico upgraded to 7.2 magnitude, from 6.9 - USGS
  (Eguneraketa: Baja California, Mexikoko lurrikararen magnitudea 6.9tik 7.2ra eguneratuta - USGS)

Ezagutza-basea aztertu ondoren (2 taula), honela etiketatzen da urruneko gainbegiraketaren bidez:

- Update : Earthquake in <region>Baja California< /region> , <country>Mexico< /country> upgraded to <magnitude>7.2< /magnitude> magnitude , from 6.9 - USGS

Guztira 13562 aipamen etiketatu ditu UG sistemak. 2 taularen azken zutabean aurki dezakegu argumentu bakoitza zenbat aldiz etiketatu den txioen datu-multzoan.

3.4 Argumentuen kategorizazioa ikasketa automatikoarekin

Gure sistemak, nolabait esateko, *burmuin* bat dauka integratuta, **sailkatzaile** deiturikoa. Sailkatzailearen bidez informazioa kudeatzeko teknikari **ikasketa automatikoa** deitzen zaio. Ikasketa automatikoa bi fasetan dago banatuta:

- **Entrenamendu fasea:** sailkatzailearen eginbeharra etiketatutako txio guztien egitura ikastea da, txioetako elementuen ezaugarri linguistikoak aztertuz, informazio horren eredu bat sortzeko.
- **Iragarpen fasea:** sailkatzaileak beste lurrikara batzuei buruzko txioak jasotzen ditu, etiketatu gabe. Honek ikasitakoa praktikan jarri eta txioetatik informazio garrantzitsua eskuratzen du.

Sailkatzailea entrenatzeko, txio bakoitzaren ezaugarri linguistikoak behar ditugu, sailkatzaileak hauetatik ikas dezan. Horretarako, txioak tokenizatu ditugu, beste era batera esanda, hitzen banaketa bat egin, eta hitz bakoitzaren lema, kategoria gramatikala eta entitate-izen mota eskuratu. Ezaugarri linguistikoen sorkuntza Stanfordeko CoreNLP tresnaren[8] bidez egin dugu.

3 taulak aurretik jarri dugun txioaren ezaugarriak irudikatzen ditu, lehenengo zutabeak txioko hitza adierazten du, eta beste zutabeetan hitz bakoitzaren lema, kategoria gramatikala, entitate-izen mota eta kategoria ageri dira. Kategoria ezagutza-baseko argumentua da.

Ikasketa automatikorako hainbat sailkatzaile desberdin aurki ditzakegu. Bakoitzak ikasketarako bere teknika dauka. Gure esperimentuetarako erabilitako sailkatzailea "Baldintzazko hausazko eremua" da (BHE, ingelesez, *Conditional Random Field*[9]). Sailkatzaile hau etiketatze sekuentzialean oinarritzen da, eta hitz bakoitzaren inguruko hitzak aztertzen ditu datu-multzoa entrenatzean, baita hitz baten etiketa iragartzean ere. Aukeratutako BHE sailkatzailea Stanfordeko CoreNLP tresnarena da.

[8] `http://nlp.stanford.edu/software/corenlp.shtml`
[9] Wikipedian: `http://en.wikipedia.org/wiki/Conditional_random_field`

3 Taula: Txio baten aurreprozesaketa. Erabilitako txioa taularen gainean dago. Txioan hitzak banandu dira eta bakoitzarentzat bere lema, kategoria gramatikala eta entitate-izen motaren balioak lortu. Azken zutabean, hitzari dagokion kategoria dago, urruneko gainbegiraketaren bidez sistemak etiketatutakoa.

Update : Earthquake in <region>Baja California</region> , <country>Mexico</country> upgraded to <magnitude>7.2</magnitude> magnitude , from 6.9 - USGS

Hitza	**Lema**	**Kat. gram.**	**Entitate-izena**	**Kategoria**
Update	Update	NNP	O	O
:	:	:	O	O
Earthquake	earthquake	NN	O	O
in	in	IN	O	O
Baja	Baja	NNP	LOCATION	region
California	California	NNP	LOCATION	region
,	,	,	O	O
Mexico	Mexico	NNP	LOCATION	country
upgraded	upgrade	VBN	O	O
to	to	TO	O	O
7.2	7.2	CD	NUMBER	magnitude
magnitude	magnitude	NN	O	O
,	,	,	O	O
from	from	IN	O	O
6.9	6.9	CD	NUMBER	O
-	-	:	O	O
USGS	usg	NN	ORGANIZATION	O

Sailkatzailea Txinako lurrikara bati buruzko adibide honetatik ahalik eta informazio gehien lortzen saiatzen da:

- Earthquake in western China kills more than 60 - The 7.1 quake struck around 33 km below the surface in Yushu county ... `http://ow.ly/173YIJ`
  (Txinako mendebaldeko lurrikarak 60 pertsona baino gehiago hil ditu - 7.1eko dardara gainazaletik 33 km-ko sakoneran talka Yushu udalerrian ... `http://ow.ly/173YIJ` *)*

Txio honen ezaugarri linguistikoak aztertu ondoren, gai izan beharko litzateke ondorengo informazioa erauzteko:

Argumentua	**Estatua**	**Herrialdea**	**Hildakoak**	**Magnitudea**	**Sakonera**
Balioa	Txina	Yushu	60 baino gehiago	7.1	33 km

Sailkatzaleak lurrikara konkretu baten txio berri guztiak aztertu ondoren, argumentu bakoitzarentzat iragarpen desberdinak egiten ditu, baina argumentu gehienek balio bakarra onartzen da. Iragarpen egokiena aukeratzeko, *NoisyOR* metodoa erabili dugu. *NoisyOr* egokia da kategorizaziorako, ereduen konfidantza (zenbat eta probabilitate altuagoa, orduan eta puntuazio altuagoa) eta jarioa (zenbat eta aipamen gehiago etiketa baterako iragarri, orduan eta altuagoa izango da etiketaren puntuazioa) ondo orekatzen dituelako:

$$NoisyOr(a,i) = 1 - \prod_{p \in P} (1-p) \tag{1}$$

a argumentuaren izena da eta i argumentuaren iragarpen potentziala. Txio bakoitzean, sailkatzaileak iragarpen-probabilitate bat (p) ematen dio hitz bakoitzari argumentu bakoitzeko. P aldagaiak iragarpen-probabilitate guztiak multzokatzen ditu, a argumenturako. Formula hau (Surdeanu *et al.*, 2012) lanetik hartu dugu.

4 Taula: Ebaluazioaren emaitzak.

Sistema	Doitasuna	Estaldura	F1-neurria
UG	50.60	17.79	24.07
Eskuzkoa	47.65	26.69	34.21

3.5 Emaitzak

Entrenamendurako, ezagutza-baseko lurrikaren %75a erabili dugu, gainentzekoa ebaluaziorako.

Gure sistemaren emaitzak ebaluatzeko erabili ditugun ebaluazio-metrikak doitasuna, estaldura, eta F1-neurria dira.

Doitasunak sistemak itzulitako emaitza zuzenen kopurua itzulitako guztiekin konparatzen du:

$$Doitasuna = \frac{\#(Emaitza\_zuzenak)}{\#(Sistemak\_itzulitako\_emaitzak)} \tag{2}$$

Estaldurak sistemak itzulitako emaitza zuzenen kopurua EBan dauden balio guztiekin konparatzen du:

$$Estaldura = \frac{\#(Emaitza\_zuzenak)}{\#(Asmatu\_behar\_direnak)} \tag{3}$$

Eta **F1-neurria** doitasuna eta estalduraren arteko batazbesteko harmonikoa da:

$$F1Neurria = 2 * \frac{doitasuna * estaldura}{doitasuna + estaldura} \tag{4}$$

Gure sistemaren eraginkortasuna ondo neurtzeko, txio guztiak eskuz etiketatu ditugu, eta ikasketa automatikoa aplikatu, aurretik aipatu dugun metodologia erabiliz. Eskuzko etiketatzearen bidez, gure sistemak lortuko lukeen emaitza onena kalkulatu dezakegu, eta UG sistemak lortutakoarekin konparatu.

4 taulan ikus ditzakegu UG algoritmoaren eta eskuzko etiketatzearen ebaluazioen emaitzak. Gure sistemak estaldura txikiagoa dauka, baina eskuzkoak baino doitasun hobea. F1-neurritik ez gaude urruti.

Ikusten den bezala, urruneko gainbegiraketak potentzial handia dauka gertaeren erauzketarako. Mikroblogak erabiltzea informazioa lortzeko eraginkorra dela frogatu dugu ere.

4 Ondorioak

Artikulu honetan Twitterreko txioetatik lurrikarei buruzko informazioa eskuratzeko sistema bat aurkeztu dugu. Horretarako, urruneko gainbegiraketaren (UG) algoritmoa gertaera erauzketarako moldatu dugu. UG algoritmoak corpusak automatikoki etiketatzen ditu, eskuzko lan garestia ekidituz. Lan honetan frogatzen dugu posible dela UG gertaera erauzketarako ere aplikatzea. Esperimentuetarako aukeratutako domeinua lurrikarena da.

Lan honetan mikroblogetatik gertaerei buruzko informazioa eskuratzea posible dela frogatzen dugu, gertaera erauzketan hauen potentziala argi utziz, nahiz eta hauek hizkera kolokiala, sintaxi zaratatsua eta informazio anbiguoa eduki.

Gure esperimentuetarako lurrikarei buruzko ezagutza-base bat sortu dugu. Horrez gain, ezagutza-basean dauden lurrikara desberdinei buruzko txioak eskuratu ditugu Twitterretik, gure esperimentuetan erabiltzeko. Ezagutza-basea eta txioen datu-multzoa publikoki eskuragarri daude.

Gure sistemaren eraginkortasuna neurtzeko, eta lortuko lukeen emaitza maximoa jakiteko, txioak eskuz etiketatu genituen. Txio hauen entrenamenduak UGrekin erabilitako txioen urrats berdinak jarraitzen ditu. Guk lortutako emaitzak eskuzkoaren emaitzetatik gertu daude, UG algoritmoak mikroblogetatik gertaerak erauzteko duen gaitasuna frogatuz.

5 Etorkizuneko lanak

Txioak eskuz etiketatzean, hauetan aurki ditzakegun datuak oso dinamikoak direla konturatu gara. Horrez gain, sistemak iragarritako balio asko ezagutza-basekoen oso hurbilak zirela ere. Txioetako informazioa EBkoekiko antzekoa denean, hauek ere etiketatzen egingo ditugu esperimentuak, emaitzak hobetzeko. Ebaluazioan antzeko balioak partzialki ontzat hartzea ere lan polita litzakete.

Sarreran aipatu bezala, txioetan topatu dezakegun informazioa oso anbiguoa da. Anbiguotasun horri aurre egiteko asmoa dugu, lurrikara bereko txioen artean hauen testuingurua elkarbanatuz.

Lan honetan esperimentuak domeinu bakar baterako bakarrik egin ditugu, eta komeni zaigu beste domeinutan frogak egitea. (Reschke *et al.*, 2014) lanerako, hegazkin-istripuei buruzko ezagutza-base bat sortu zuten; EB hau aprobetxatu dezakegu Twitterretik istripu hauei buruzko txio desberdinak eskuratu eta gure esperimentuak errepikatzeko.

Amaitzeko, Interesgarria litzateke gure sistema moldatzea informazioa Twitterretik denbora errealean eskuratzeko.

Erreferentziak

Benson, Edward, Aria Haghighi, eta Regina Barzilay. 2011. Event discovery in social media feeds. In *Proceedings of the Conference of the Association for Computational Linguistics (ACL)*.

Intxaurrondo, Ander, 2015 (Depositatzeko). *Ezagutza-baseen aberasketa urruneko gainbegiraketaren bidez: analisiak eta hobekuntzak*. Euskal Herriko Unibertsitatea tesia.

Mintz, Mike, Steven Bills, Rion Snow, eta Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics*.

Reschke, Kevin, Martin Jankowiak, Mihai Surdeanu, Christopher D. Manning, eta Daniel Jurafsky. 2014. Event extraction using distant supervision. In *Proceedings of LREC*.

Surdeanu, Mihai, Julie Tibshirani, Ramesh Nallapati, eta Christopher D. Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing and Natural Language Learning (EMNLP-CoNLL)*.

6 Eskerrak eta oharrak

Arizonako Unibertsitateko Mihai Surdeanu ikerlariari eskerrak eman nahi dizkiogu, lan honetan eman digun laguntzagatik.

Esker instituzionalak Eusko Jaurlaritzako Hezkuntza, Unibertsitate eta Ikerketa Sailari, ikerketa lan hau egiteko emandako ikertzaileak prestatzeko bekarengatik.

Lan hau egile nagusiaren tesiaren eratorria da (Intxaurrondo, 2015 (Depositatzeko)).

Irudiaren iturria: `http://zthiztegiberria.elhuyar.org/artikuluak/Lurrikara`