Linguistic Redundancy in Twitter

Fabio Massimo Zanzotto
University of Rome "Tor Vergata"
Rome, Italy
zanzotto@info.uniroma2.it

Marco Pennacchiotti
Yahoo! Labs
Sunnyvale, CA, 94089
pennac@yahoo-inc.com

Kostas Tsioutsiouliklis
Yahoo! Labs
Sunnyvale, CA, 94089
kostas@yahoo-inc.com

Abstract

In the last few years, the interest of the research community in micro-blogs and social media services, such as Twitter, is growing exponentially. Yet, so far not much attention has been paid on a key characteristic of micro-blogs: the high level of information redundancy. The aim of this paper is to systematically approach this problem by providing an operational definition of redundancy. We cast redundancy in the framework of Textual Entailment Recognition. We also provide quantitative evidence on the pervasiveness of redundancy in Twitter, and describe a dataset of redundancy-annotated tweets. Finally, we present a general purpose system for identifying redundant tweets. An extensive quantitative evaluation shows that our system successfully solves the redundancy detection task, improving over baseline systems with statistical significance.

1 Introduction

Micro-blogs and social media services, such as Twitter, have experienced an exponential growth in the last few years. The interest of the research community and the industry in these services has followed a similar trend. Web companies such as Google, Yahoo, and Bing are integrating more and more social content to their sites. At the same time, the computational linguistic community is getting increasingly interested in studying social and linguistic properties of Twitter and other micro-blogs (Java et al., 2007; Krishnamurthy et al., 2008; Kwak et al., 2010; Zhao et al., 2007; Popescu and Pennacchiotti, 2010; Petrović et al., 2010; Lin et al., 2010; Liu et al., 2010; Ritter et al., 2010). Yet, so far, not much attention has been paid on a key characteristic of micro-blogs: the high level of information redundancy. Users often post messages with the same, or very similar, content, especially when reporting or commenting on news and events. For example, the following two tweets are part of a large set of redundant tweets issued during the 2010 winter Olympics:

(example 1)

t_1 : *"Swiss ski jumper Simon Ammann takes first gold of Vancouver"*

t_2 : *"Swiss (Suisse) get the Gold on Normal Hill ski jump. #Vancouver2010"*

By performing an editorial study (described later in the paper) we discovered that a large part of event-related tweets are indeed redundant.

Detecting information redundancy is important for various reasons. First, most applications based on Twitter share the goal of providing tweets that are both *informative* and *diverse*, with respect to an initial user information need. For example, Twitter search engines should ideally select the most informative and diverse set of tweets in return to a user query. Similarly, a news web portal that attaches tweets to a given news article should attach those tweets that provide the broadest and most diverse set of information, opinions, and updates about the news item. To keep a high level of diversity, redundant tweets should be removed from the set of tweets displayed to the user. Figure 1 shows an example of a Twitter search engine where redundant tweets are

Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 659–669,
Edinburgh, Scotland, UK, July 27–31, 2011. ©2011 Association for Computational Linguistics

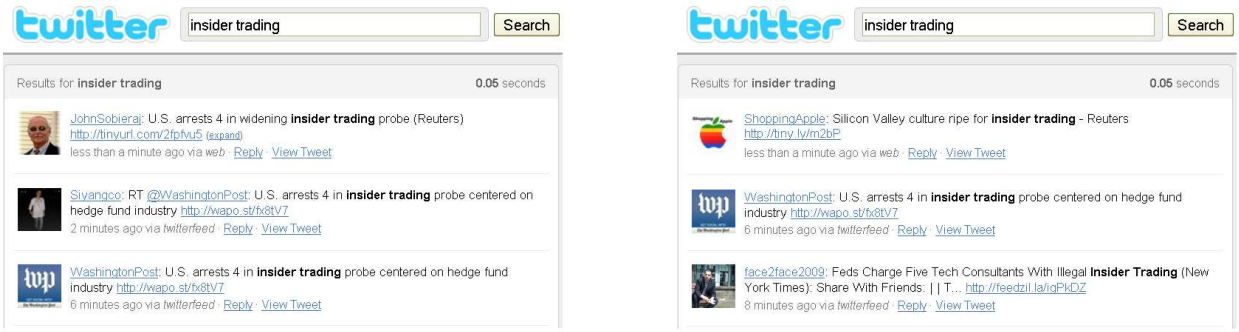


Figure 1: Twitter search: actual Twitter results and desired results after redundancy reduction.

present (left) and where they are discarded (right).

Also, from a computational linguistic point of view, the high redundancy in micro-blogs gives the unprecedented opportunity to study classical tasks such as text summarization (Haghighi and Vanderwende, 2009), textual entailment recognition (Dagan et al., 2006) and paraphrase detection (Dolan et al., 2004) on very large corpora characterized by an original and emerging linguistic style, pervaded with ungrammatical and colloquial expressions, abbreviations, and new linguistic forms.

The aim of this paper is to formally define, for the first time, the problem of redundancy in micro-blogs and to systematically approach the task of automatic redundancy detection. Note that we focus on linguistic redundancy, i.e. tweets that convey the same information with different wordings, and ignore the more trivial issue of detecting retweets, which can be considered the most basic expression of redundancy.

The main contributions of this paper are the following:

- We formally define the problem of redundancy detection in micro-blogs within the framework of Textual Entailment theory;
- We report results from an editorial study and provide quantitative evidence of the pervasiveness of redundancy in Twitter;
- We present a set of simple and effective machine learning models for solving the task of redundancy detection;
- We provide promising experimental results that show that these models outperform baseline approaches with statistical significance, and we report a qualitative evaluation revealing the advantages of the proposed model.

The rest of the paper is organized as follows. First, we shortly describe related work in Section 2. Next, we provide our operational definition of redundancy and introduce our editorial study and dataset in Section 3. In Section 4 we describe our models for redundancy detection. In Section 5 we provide a quantitative and qualitative evaluation of our models. In Section 6 we conclude the paper with final observations and future work.

2 Related Work

So far, most research on **Twitter** has focused on its network structure, the social behavior of its users (Java et al., 2007; Krishnamurthy et al., 2008; Kwak et al., 2010), ranking tweets by relevance for web search (Ramage et al., 2010; Duan et al., 2010), and the analysis of time series for extracting trending news, events and facts (Zhao et al., 2007; Popescu and Pennacchiotti, 2010; Petrović et al., 2010; Lin et al., 2010). Only few studies have specifically focused on the linguistic content analysis of tweets, e.g. (Davidov et al., 2010; Barbosa and Feng, 2010). To date, our paper most closely relates to works on semantic role labeling (SRL) on social media (Liu et al., 2010) and conversation modeling (Ritter et al., 2010).

Liu et al. (2010) present a self-learning SRL system for news tweets, with the goal of addressing low performance caused by the noise and the unstructured nature of the data. The authors first cluster together tweets that refer to the same news. Then, for each cluster, they identify the tweets that are

well-formed (i.e. copy-pasted from news), and induce role mappings between well-formed and noisy tweets in the same cluster by performing word alignment. In our paper we are also interested in aligning and grouping tweets, although our goal is to detect redundancy, not to perform SRL.

On a different ground, Ritter et al. (2010) propose a probabilistic model to discover dialogue acts in Twitter conversations and to classify tweets in a conversation according to those acts. (A conversation is defined as a set of tweets in the same reply thread.) The authors define 10 major dialogue acts for Twitter, including status, question, response and reaction, and automatically build a probabilistic transition graph for such acts. In our paper, we also aim at classifying tweets, but our interest is in information redundancy instead of acts.

In the computational linguistic literature, **redundancy detection** is studied in multi-document summarization, where the overall document is used to select the most informative sentences or snippets (Haghighi and Vanderwende, 2009). Since tweets are short and tweet sets cannot be considered documents, these methods are hard to apply. A more convenient setting is paraphrase detection (Dolan et al., 2004) and textual entailment recognition (Dagan et al., 2006) (RTE).

In RTE the task is to recognize if a text called the *text* T (typically one or two sentences long) entails another text called the *hypothesis* H. Many approaches have been proposed for this task, mostly based on machine learning. Three main classes of features have been so far explored in RTE: distance/similarity feature spaces (Corley and Mihalcea, 2005; Newman et al., 2005; Haghighi et al., 2005; Hickl et al., 2006), entailment trigger feature spaces (de Marneffe et al., 2006; MacCartney et al., 2006), and pair content feature spaces (Zanzotto et al., 2009). Distance/similarity feature spaces are more suitable to the paraphrase detection task because they model the similarity between the two texts. On the other hand, entailment trigger and content feature spaces model complex relations between the texts, taking into account first-order entailment rules, i.e. entailment rules with variables.

In this paper, one of our goals is to explore RTE techniques and features that are usually used for classical texts, and check if they can be successfully adapted to the unstructured, and oftentimes ungrammatical, Twitter language.

3 Redundancy in Twitter

We formally define two tweets as **redundant** if they either convey the same information (*paraphrase*) or if the information of one tweet subsumes the information of the other (*textual entailment*). For example, the pair in *(example 1)* is redundant. The first tweet subsumes (i.e. 'textually entails') the other; both tweets state that Switzerland won a Gold Medal at the Vancouver winter Olympics, but the first one also specifies the name of the athlete. The following pair is, instead, non-redundant, because the two tweets convey different information, and they do not subsume each other:

(example 2)

t_1 : *"Goal! Iniesta scores for #ESP and they have one hand on the #worldcup"*

t_2 : *"this will be a hard final #Esp vs Ned #worldcup"*

Our definition of redundancy is grounded on, and inspired by, the theory of Textual Entailment, to which we refer the reader for further details (Dagan et al., 2006).

3.1 Quantifying redundancy

How pervasive is redundancy in Twitter? In order to answer this question we performed an initial editorial study where human editors were asked to annotate pairs of tweets as being either redundant or non-redundant. The editorial study also serves as a test bed for evaluating our redundancy detection models, as discussed in Section 5.

In the study we focus on 'informative' tweets, i.e. tweets that describe or comment on relevant events/facts. Indeed, these are the types of tweets for which redundancy is a critical issue, especially in view of real applications, e.g. to present a diverse set of tweets for a given news article. Other types of tweets, such as status updates, self-promotions, and personal messages are of less interest in this context.

Dataset extraction. The study is performed on an automatically built dataset of informative tweets. The most critical issue for extracting the dataset is to pre-process tweets and to discard those that are

not informative. This is not an easy task: a recent study (Pear-Analytics, 2009) estimates that only 4% of all tweets are factual news, and only 37% are conversations with content. The rest are spam, status updates and other types of uninformative content. In order to retain only informative tweets we first extract *buzzy snapshots* (Popescu and Pennacchiotti, 2010). A snapshot is defined as a set of tweets that explicitly mention a specific topic within a specified time period. A buzzy snapshot is defined as a snapshot with a large number of tweets, compared to previous time periods. For example, given the topic 'Haiti earthquake', the snapshot composed by the tweets mentioning 'Haiti earthquake' on January 12th, 2010, will constitute a buzzy snapshot, since in previous days the topic was not mentioned often.

We use two different topic lists: a *celebrity list* containing about 104K celebrity names, crawled from Wikipedia, including actors, musicians, politicians, and athletes; and an *event list* composed of 398 hashtags related to 8 major events that happened between January and July 2010, and listed in Wikipedia: [1] the earthquake in Haiti, the winter Olympics, the earthquake in Chile, the death of the Polish president, the volcano eruption in Iceland, the oil spill in the Gulf of Mexico, the Greek financial crisis, and the FIFA world cup.

We extract buzzy snapshots for the above two topic lists by following the method described in (Popescu and Pennacchiotti, 2010): we consider time periods of one day, and call buzzy the snapshots that mention a given topic α times more than the average over the previous 2 days. We set α to 20 and 5 respectively for the celebrity list and the event list. We further exclude irrelevant and spam snapshots by removing those that have: fewer than 10 tweets; more than 50% of tweets non-English; and an average token overlap between tweets of more than 80%, usually corresponding to spam threads.

The extraction is performed on a Twitter corpus containing all tweets posted between July 2009 and August 2010. In all, we extract 972 snapshots for the celebrity list, containing 205,885 tweets (i.e. average of 212 tweets per snapshot); and 674 snapshots for the event list, containing 393,965 tweets (584 tweets per snapshot).

[1] Hashtags are keywords prefixed by '#', that are used by the Twitter community to mark the topic of a tweet. We collected our set of hashtags by semi-automatically inspecting the Twitter stream in the days the major events happened.

redundant	**367**	**(29.5%)**
entailment	195	(15.7%)
paraphrase	172	(13.5%)
non-redundant	**875**	**(70.5%)**
related	541	(43.6%)
unrelated	334	(26.9%)

Table 1: Results of the redundancy editorial study.

The above two final snapshot corpora (i.e. the 972 celebrities' snapshots and 674 events' snapshots) can be considered a good representation of event descriptions and comments on Twitter, thus forming our initial set of 'informative' tweets. From these two corpora, we extract the final tweet-pair dataset by randomly sampling 1500 pairs of tweets contained in the same snapshot. Tweet-pairs that contain retweets are excluded.

Dataset annotation. The main editorial task consisted of annotating tweet-pairs as either redundant or non-redundant. We also asked editors to characterize the specific linguistic relation between the two tweets of a pair. We consider four relations: *entailment* (the first tweet entails the second or vice versa), *paraphrase*, *contradiction* (the tweets contradict each other), and *related* (the tweets are about the same topic, e.g. the Haiti earthquake, but are in none of the previous relations). Tweets that were about different topics were labeled *unrelated*. Annotators were asked to base their decisions on the parts of the tweets that contained information relevant to the selected topic, e.g. the earthquake in Haiti. These parts were marked in the corpus. Focusing on these parts is in line with potential applications of tweet redundancy detection as tweets are firstly grouped around a topic. Note that pairs that fall under the entailment or paraphrase relation are redundant, while unrelated, related, and contradictory tweets are non-redundant.

The annotation was performed in a three stage process, since tweets are sometimes hard to understand and hence to annotate (misspellings, usage of slang and abbreviations, lack of discourse context). In the first step, the 1500 pairs were independently annotated by a pool of 20 trained editors, super-

vised by an expert lead. In the second step, the annotations were checked by three highly trained experts with background in computational linguistics: each pair was independently checked by two experts. Average kappa agreement in this second step is $kappa = 0.63$ (corresponding to 'good agreement'). In a final step, discordances between the two experts were resolved by the third expert. Unclear and unresolved pairs after the three stages were discarded from the dataset, leaving a final set of 1242 pairs. [2]

Annotation Results. Table 1 reports the results of our study. Among the 1242 tweet-pairs, 367 (30%) are redundant and 875 (70%) are non-redundant. This shows that redundancy is indeed a pervasive phenomenon in Twitter, and a critical issue that has to be solved in order to provide clean and diverse social content. Most cases of redundancy correspond to tweets that report the same fact using different wording, occasionally adding irrelevant personal comments and sentiments (e.g. 'Johnny Depp died' vs. 'OMG, I am so sad that Johnny Depp is dead').

4 Redundancy detection models

The task of **redundancy detection** in Twitter is a tweet-pair classification problem. Given two tweets t_1 and t_2, the goal is to classify the pair (t_1, t_2) as being either redundant or non-redundant.

In this section we describe different models for redundancy detection, inspired by existing work in RTE. We adopt a machine learning approach where a Support Vector Machine (SVM) is trained on a manually annotated training set to classify incoming test examples as either redundant or non-redundant. An evaluation of the different models adopting for training and testing the dataset described in Section 3, is presented in Section 5.

4.1 Bag-of-word model (BOW)

The bag-of-word model is the most simple approach for detecting redundancy. It is used as a **baseline** in our experiment. The simple intuition of the model is that if two tweets t_1 and t_2 have a high lexical overlap, then they are likely to express the same information – i.e. they are likely to be redundant. In this model, the SVM is trained using a single feature that computes the cosine similarity between the bag-of-word vectors of the two tweets. The bag-of-word vector is built using a classical *tf*idf* weighting schema over the set of tokens of the pair. This a very simple baseline as SVM is only learning thresholds using this single feature.

The bag-of-word model is of course a naive approach, since in many cases redundant tweets can have very different lexical content (e.g. the following two tweets: "Farrah Fawcett left out of Oscar memorial", "No Farrah Fawcett's memory at the Academy Awards"), and non-redundant tweets can have similar lexical content (e.g. the tweets: "Johnny Deep is dead", "Johnny Deep is not dead").

4.2 WordNet-based bag-of-word model (WBOW)

The second **baseline** model was first defined in (Corley and Mihalcea, 2005) and since then has been used by many RTE systems. The model extends BOW by measuring similarity at the semantic level, instead of the lexical level.

For example, consider the tweet pair: "Oscars forgot Farrah Fawcett", "Farrah Fawcett snubbed at Academy Awards". This pair is redundant, and, hence, should be assigned a very high similarity. Yet, BOW would assign a low score, since many words are not shared across the two tweets. WBOW fixes this problem by matching 'Oscar'-'Academy Awards' and 'forgot'-'snubbed' at the semantic level. To provide these matches, WBOW relies on specific word similarity measures over WordNet (Miller, 1995), that allow synonymy and hyperonymy matches: in our experiments we specifically use Jiang&Conrath similarity (Jiang and Conrath, 1997).

In practice, we implement WBOW by using the text similarity measure defined in (Corley and Mihalcea, 2005) as the single feature in the SVM classifier that, as in BOW, learns the threshold on this single feature.

4.3 Lexical content model (LEX)

This model and the next ones (SYNT and FOR) explicitly model the content of a tweet pair $P =$

[2] At this time, the Twitter^{TM} Terms of Use do not allow publication of the annotated dataset. Should the Terms of Use change, the dataset will become available for download at *http://art.uniroma2.it/zanzotto/datasets*.

(t_1, t_2) as a whole. This is a radically different approach with respect to the similarity-based models explored so far, where the content of t_1 and t_2 were treated independently (i.e. each tweet with its own bag of words), and the SVM used as the single feature the similarity between the two tweets.

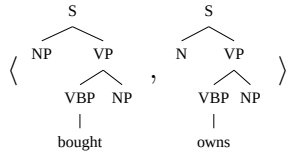
In the LEX model we represent the content of the tweet pair in a double bag-of-word vector space. Each pair $P = (t_1, t_2)$ is represented by two bag-of-word vectors, $(\vec{t_1}, \vec{t_2})$. Within this space, we can then define a specific similarity measure between pairs using a kernel function in the SVM learning algorithm. Given two pairs of tweets $P^{(a)}$ and $P^{(b)}$, the LEX kernel function is defined as follows:

$$K_{LEX}(P^{(a)}, P^{(b)}) = cos(t_1^{(a)}, t_1^{(b)}) + cos(t_2^{(a)}, t_2^{(b)})$$

where $cos(\cdot, \cdot)$ is the cosine similarity between the two vectors. The LEX feature space is simple and can be extremely effective in modeling the content of tweet pairs. Yet, in principle, it doesn't model the relations among words in the tweet. Different content feature spaces are then needed to capture these relations.

4.4 Syntactic content model (SYNT)

The SYNT model represents a tweet pair using pairs of syntactic tree fragments from t_1 and t_2. Each feature is a pair $< fr_1, fr_2 >$, where fr_1 and fr_2 are syntactic tree fragments (see figure below). As defined in (Collins and Duffy, 2002), a syntactic tree fragment fr_i is active in t_i when fr_i is a subtree of the syntactic interpretation of t_i. Therefore, these features represent ground rules connecting the left-hand sides and the right-hand sides of the tweet pair: each feature is active for a pair (t_1, t_2) when the left-hand side fr_1 is activated by the syntactic analysis of t_1 and the right-hand side fr_2 is activated by t_2. As an example consider the feature:



This feature is active for the pair of tweets ("*GM bought Opel*","*GM owns Opel*") since the syntactic analysis of the pair matches the feature (given that the two tweets are correctly syntactically analyzed). This feature space models the relations between words syntactically. Therefore it overcomes the limitations of the LEX feature space. But it also introduces a new limitation: the above feature is in fact also active for the tweet pair ("*GM bought Opel*","*Opel owns GM*"). This pair is extremely different from the previous one, thus possibly misleading the classifier.

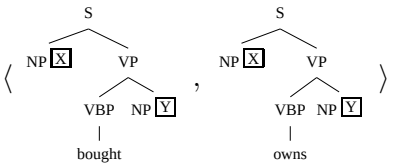
This feature space is not represented explicitly, but it is encoded in a kernel function. Given two pairs of tweets $P^{(a)}$ and $P^{(b)}$, the SYNT kernel function is defined as follows:

$$K_{SYNT}(P^{(a)}, P^{(b)}) = K(t_1^{(a)}, t_1^{(b)}) + K(t_2^{(a)}, t_2^{(b)})$$

where $K(\cdot, \cdot)$ is the tree kernel function described in (Collins and Duffy, 2002).

4.5 Syntactic first-order rule content model (FOR)

The FOR model overcomes the limitations of SYNT, by enriching the space with features representing first-order relations between the two tweets of a pair. Each feature represents a rule with variables, i.e. a first order rule that is activated by the tweet pairs if the variables are unified. This feature space has been introduced in (Zanzotto and Moschitti, 2006) and shown to improve over the ones above. Each feature $< fr_1, fr_2 >$ is a pair of syntactic tree fragments augmented with variables. The feature is active for a tweet pair (t_1, t_2) if the syntactic interpretations of t_1 and t_2 can be unified with $< fr_1, fr_2 >$. For example, consider the following feature:



This feature is active for the pair ("*GM bought Opel*","*GM owns Opel*"), with the variable unification X = "*GM*" and Y = "*Opel*". On the contrary, this feature is not active for the pair ("*GM bought Opel*","*Opel owns GM*") as there is no possibility of unifying the two variables. Efficient algorithms for the computation of the related kernel functions can

be found in (Moschitti and Zanzotto, 2007; Zanzotto and Dell'Arciprete, 2009).

5 Experimental Evaluation

In this section we present an evaluation of the different redundancy detection models. First, we define the experimental setup in Section 5.1. Then, we analyze the results of the experiments in Section 5.2.

5.1 Experimental Setup

We experiment with the redundancy detection dataset described in Section 3. We randomly divide the corpus into two sets: 50% for training and 50% for testing. The training set contains 185 positive tweet-pairs and 416 negative pairs. The test set contains 182 positive pairs and 466 negatives.

We evaluate the performance of the SVM models using the following feature combinations: LEX+BOW, LEX+WBOW, SYNT+BOW, SYNT+WBOW, FOR+BOW, FOR+WBOW. We compare to the system baselines BOW and WBOW. [3]

The performance of the different models is computed using the Area Under the ROC curve (AROC) applied to the classification score returned by the SVM. The ROC curve allows us to study the behavior of the classifier in detail, and also provides a powerful way to compare among systems when the dataset is unbalanced (as in our case).

To determine the statistical significance of the difference in the performance of the systems we analyzed, we use the model described in (Yeh, 2000) as implemented in (Padó, 2006).

We pre-process the dataset with the following tools: the Charniak Parser (Charniak, 2000) for parsing sentences, the WordNet similarity package (Pedersen et al., 2004) for computing WBOW and for linking the two tweets in a pair, and SVM-light (Joachims, 1999), extended with the syntactic first-order rule kernels described in (Moschitti and Zanzotto, 2007) for creating the SYNT and the FOR feature spaces. We used the Charniak syntactic parser without any specific adaptation to the Twitter language.

Model	AROC
BOW	0.592
WBOW	0.578
LEX + BOW	0.725 †
LEX + WBOW	0.728 †
SYNT + BOW	0.736 †
SYNT + WBOW	0.737 †
FOR + BOW	0.739 †
FOR + WBOW	**0.747** † ‡

Table 2: Experimental results of the different systems. † indicates statistical significance ($p < 0.01$) with respect to the two baseline methods BOW and WBOW. ‡ indicates statistical significance ($p < 0.1$) with respect to FOR + BOW

5.2 Experimental Results

Table 2 reports the results of the experiment. The first and most important result is that models using content features (LEX, SYNT, and FOR) along with similarity features (BOW and WBOW) outperform the two baseline models using only similarity features with statistical significance, up to more than 15% AROC points.

At first glance, WordNet similarities are not useful: the performance of the WBOW model is indeed comparable and statistically insignificant with respect to the pure token based model BOW. This seems to be intuitive as the language of the tweets can be far from proper English, i.e. it may contain many out-of-dictionary words that are not present in WordNet, thus impairing the similarity measure used by WBOW.

This trend is also confirmed in the case of content-based systems like LEX and SYNT. Using BOW or WBOW in combination with these features has the same effect on the final performance. Only the FOR features are positively affected by the WordNet-based distance. This may be explained by the fact that in the FOR+WBOW system, the WordNet similarity is also used to link words in the two tweets of a pair. This increases the possibility of finding reasonable and useful first-order rules. In the quali-

[3] Note that other feature combinations would not add value, as BOW and WBOW are interchangeable, and the same stands for LEX, SYNT and FOR.

tative analysis that follows, we show some examples that support this intuition.

On the other hand, syntax plays a key role for detecting redundancy. The two syntax based models SYNT and FOR outperform the lexical based models LEX between 1 and 2 AROC points. This is surprising, since the Charniak parser used in the experiments has not been adapted to the Tweet language, and therefore could have produced many interpretation errors, thus impairing the use of syntax. This seems to suggest that if the interpretations of the part-of speech tags of the unknown words is correct, the syntax of tweets is reasonably similar to the syntax of the generic English language.

The best performing model is FOR+WBOW: first-order rules successfully emerge in tweets and are positively exploited by the learning system. In the next section we report examples that support this observation.

5.3 Qualitative analysis

The experimental results reported in the previous section show that first-order syntactic rules in combination with the WordNet-based bag-of-word (FOR+WBOW) are highly effective in detecting redundancy. In this section, we briefly analyze some tweet pairs where the differences between this model and the BOW and WBOW models are evident.

Table 3 reports examples of tweet pairs, along with their ranking position in the test set, according to the SVM score, with respect to different models. The first column represents the editorial gold standard (gs) for the tweet pairs we considered: either redundant (R) or non-redundant (N). Since we feed the classifiers with 'redundant' as the positive class [4], a classifier is better than another if it ranks redundant tweet pairs (R) higher than non-redundant ones (N). The second, the third, and the fourth columns represent the rank given by WBOW+FOR, WBOW, and BOW respectively. The fifth column is the tweet-pair identifier in our dataset (id). The last two columns are the two tweets in each pair.

The table reports interesting examples where redundant pairs have very little lexical similarity while the non-redundant pairs have a high lexical similarity. These are all examples where BOW and WBOW should typically fail, while FOR+WBOW could capture important syntactic first-order rules to overcome the limitations of the pure similarity-based models.

As a first example, both BOW and WBOW fail to assign a high rank (i.e. low rank number) to the redundant pair *o165*: in fact, 'died' does not lexically match 'rip', nor are these two words related in WordNet. In contrast, FOR+WBOW assigns a high rank to this pair, since it may be able to apply the rule $<$*X died*, *rip X*$>$ that was most probably acquired from examples in the training set (the hoax of somebody's death is pervasive in Twitter, and it is therefore likely to fire the abovementioned rule in our dataset if enough examples are available).

The third and the fourth pairs (*o130* and *o21*) show some commonalities [5] . According to the WordNet similarity measure we used, 'recognize' and 'snub' are highly related as well as 'forget' and 'snub'. Hence, the two tokens are linked as similar. For *o130*, the triggering syntactic rule is $<$(S (NP *X*) (VP *Y*),(VP (V *Y*) (NP *X*)$>$ where *X* and *Y* are variables. For *o21*, the rule is: $<$(VP (V *X*) (NP *Y*),(VP (V *X*) (NP *Y*)$>$.

For the non-redundant pairs (N) at the bottom of the table, the first-order rules are less intuitive. Yet, it is clear why these pairs have high lexical similarity (and therefore are ranked high by BOW and WBOW): The two tweets in the pair *oe387* share 'volcanic', 'ash', and the hashtag '#ashtag'. Tweets in *oe64* share 'Icelandic' and 'eruption' but they are describing different facts. Tweets in the pair *oe43* are similar since they are sharing the three hashtags '#bpoil', '#bp', and '#oilspill'. This example shows that hashtags alone are not very indicative and useful for detecting redundancy in Twitter.

6 Conclusions

In this paper we introduced the notion of linguistic redundancy in micro-blogs and the task of tweet redundancy detection. We also presented an editorial study showing that redundancy is pervasive in Twitter, and that methods for its detection will be key in

[4] This is just a convention. Results would be the same by taking non-redundant pairs as the positive class.

[5] In o130, the common topic is 'farrah fawcett': "farrah fawcett not recognized at the Oscars memorial?" and "snubbed farrah fawcett. #oscars" are used by the annotators to make the decision.

gs	FOR+WBOW	WBOW	BOW	id	t_1	t_2
R	11	137	130	*o165*	"is that True that johnny depp died???"	"Rip johnny depp? This cannot be True"
R	32	246	239	*o942*	"sad...jim carrey and jenny mccarthy have called it quits..."	"jim carrey & jenny mccarthy broke up! omg! bummer! they were the cutest crazy couple ever."
R	43	165	158	*o130*	"farrah fawcett & bea arthur not recognized at the Oscars memorial? really?"	"i dont understand how they included michael jackson in the memorial tribute as an actor but snubbed farrah fawcett. #oscars"
R	101	632	641	*o21*	"Oscars forgot farrah fawcett??"	"farrah fawcett snubbed at Oscars appeared in a movie with best actor Jeff Bridges... disgusting"
N	467	161	155	*oe387*	"We may die in volcanic ash today. Choose your final pose soon to look cool for future archaeologists. #ashtag"	"# Just heard about the Icelandic volcanic ash thing, not really interested but it has the best hashtag ever, #ashtag !"
N	572	96	92	*oe43*	"Many Endangered Turtles Dying On Texas Gulf Coast http://ow.ly/1FbB8 via @nprnews #bpoil #bp #oilspill"	"Species Most at Risk Because of the Oil Spill http://ow.ly/1FcB7 #bpoil #bp #oilspill"
N	614	129	124	*oe64*	"http://bit.ly/d8W7Xw #ashtag IN PICTURES: Icelandic volcanic eruption"	"So, who's going to take a crack at pronouncing the part of Iceland the eruption was in? #ashtag"

Table 3: Ranks of some tweet pairs according to the scores of the different classifiers.

the future for the development of accurate Twitter-based applications. In the second part of the paper we presented some promising models for redundancy detection that show encouraging results when compared to typical lexical baselines. Even with the ungrammaticalities used in tweets, syntactic feature spaces are effective in modeling redundancy, especially when used in first-order rules.

In future work we plan to improve our system by adapting existing linguistic tools and resources to Twitter (e.g. syntactic parsers). We also plan to investigate the use of semantic roles and contextual information to improve the models. For example, the tweets that other users post about the same topic of the target-pair may be of some help. Finally, we are investigating the integration of our models into real applications such a the enrichment of news articles with related and *diverse* content from social media.

References

Luciano Barbosa and Junlan Feng. 2010. Robust sentiment detection on twitter from biased and noisy data. In *Posters Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 36–44, Beijing, China.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 132–139, Seattle, Washington.

Michael Collins and Nigel Duffy. 2002. New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 263–270.

Courtney Corley and Rada Mihalcea. 2005. Measuring the semantic similarity of texts. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, pages 13–18, Ann Arbor, Michigan.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The pascal recognising textual entailment challenge. In Quionero-Candela et al., editor, *LNAI 3944: MLCW 2005*, pages 177–190, Milan, Italy. Springer-Verlag.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced sentiment learning using twitter hashtags and smileys. In *Posters Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 241–249, Beijing, China.

Marie-Catherine de Marneffe, Bill MacCartney, Trond Grenager, Daniel Cer, Anna Rafferty, and Christopher D. Manning. 2006. Learning to distinguish valid textual entailments. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th International Conference on Computational Linguistics (Coling 2004)*, pages 350–356, Geneva, Switzerland.

Yajuan Duan, Long Jiang, Tao Qin, Ming Zhou, and Heung-Yeung Shum. 2010. An empirical study on learning to rank of tweets. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 295–303, Beijing, China.

Aria Haghighi and Lucy Vanderwende. 2009. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370, Boulder, Colorado.

Aria Haghighi, Andrew Ng, and Christopher Manning. 2005. Robust textual inference via graph matching. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 387–394, Vancouver, British Columbia, Canada.

Andrew Hickl, John Williams, Jeremy Bensley, Kirk Roberts, Bryan Rink, and Ying Shi. 2006. Recognizing textual entailment with LCC's groundhog system. In Bernardo Magnini and Ido Dagan, editors, *Proceedings of the 2nd PASCAL RTE Challenge*, Venice, Italy.

Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. 2007. Why we Twitter: understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007*.

Jay J. Jiang and David W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the 10th International Conference on Research in Computational Linguistics ROCLING*, pages 132–139, Tapei, Taiwan.

Thorsten Joachims. 1999. Making large-scale svm learning practical. In B. Schlkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods-Support Vector Learning*. MIT Press.

Balachander Krishnamurthy, Phillipa Gill, and Martin Arlitt. 2008. A few chirps about twitter. In *Proceedings of the first workshop on Online social networks*, pages 19–24, Seattle, WA, USA.

Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. 2010. What is twitter, a social network or a news media? In *Proceedings of WWW '10: Proceedings of the 19th international conference on World wide web*, pages 591–600, Raleigh, North Carolina, USA.

Cindy-Xide Lin, Bo Zhao, Qiaozhu Mei, and Jiawei Han. 2010. Pet: a statistical model for popular events tracking in social communities. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 929–938, Washington, DC, USA.

Xiaohua Liu, Kuan Li, Bo Han, Ming Zhou, Long Jiang, Zhongyang Xiong, and Changning Huang. 2010. Semantic role labeling for news tweets. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 698–706, Beijing, China, August.

Bill MacCartney, Trond Grenager, Marie-Catherine de Marneffe, Daniel Cer, and Christopher D. Manning. 2006. Learning to recognize features of valid textual entailments. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 41–48, New York City, USA.

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

Alessandro Moschitti and Fabio Massimo Zanzotto. 2007. Fast and effective kernels for relational learning from texts. In *Proceedings of the International Conference of Machine Learning (ICML)*, Corvallis, Oregon.

Eamonn Newman, Nicola Stokes, John Dunnion, and Joe Carthy. 2005. Textual entailment recognition using a linguistically-motivated decision tree classifier. In Joaquin Quiñonero Candela, Ido Dagan, Bernardo Magnini, and Florence d'Alché Buc, editors, *MLCW*, volume 3944 of *Lecture Notes in Computer Science*, pages 372–384. Springer.

Sebastian Padó, 2006. *User's guide to* `sigf`*: Significance testing by approximate randomisation*.

Pear-Analytics. 2009. Twitter study - august 2009.

Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. 2004. Wordnet::similarity - measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004*, pages 38–41, Boston, MA.

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

Ana-Maria Popescu and Marco Pennacchiotti. 2010. Detecting controversial events from twitter. In *In Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1873–1876.

Daniel Ramage, Susan Dumais, and Dan Liebling. 2010. Characterizing microblogs with topic models. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, pages 130–137.

Alan Ritter, Colin Cherry, and Bill Dolan. 2010. Unsupervised modeling of twitter conversations. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 172–180, Los Angeles, California.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th conference on Computational linguistics*, pages 947–953, Morristown, NJ, USA.

Fabio Massimo Zanzotto and Lorenzo Dell'Arciprete. 2009. Efficient kernels for sentence pair classification. In *Conference on Empirical Methods on Natural Language Processing*, pages 91–100, 6-7 August.

Fabio Massimo Zanzotto and Alessandro Moschitti. 2006. Automatic learning of textual entailments with cross-pair similarities. In *Proceedings of the 21st Coling and 44th ACL*, pages 401–408, Sydney, Australia, July.

Fabio Massimo Zanzotto, Marco Pennacchiotti, and Alessandro Moschitti. 2009. A machine learning approach to textual entailment recognition. *Natural Language Engineering*, 15-04:551–582.

Q. Zhao, P. Mitra, and B. Chen. 2007. Temporal and information flow based event detection from social text streams. In *Proceedings of the 22nd national conference on Artificial intelligence*, pages 1501–1506, Vancouver, British Columbia, Canada.