

Terminology Mining in Social Media

Magnus Sahlgren
SICS
Box 1263
SE-164 29 Kista, Sweden
mange@sics.se

Jussi Karlgren
SICS
Box 1263
SE-164 29 Kista, Sweden
jussi@sics.se

ABSTRACT

The highly variable and dynamic word usage in social media presents serious challenges for both research and those commercial applications that are geared towards blogs or other user-generated non-editorial texts. This paper discusses and exemplifies a terminology mining approach for dealing with the productive character of the textual environment in social media. We explore the challenges of practically acquiring new terminology, and of modeling similarity and relatedness of terms from observing realistic amounts of data. We also discuss semantic evolution and density, and investigate novel measures for characterizing the preconditions for terminology mining.

Categories and Subject Descriptors

H.3.1 [Content Analysis and Indexing]: Linguistic processing; I.2.7 [Natural Language Processing]: Text analysis; J.5 [Arts and Humanities]: Linguistics.

General Terms

Algorithms, Experimentation, Performance, Theory.

Keywords

Word Space, Distributional Semantics, Random Indexing, Terminology Mining, Social Media.

1. NEW MEDIA REQUIRE NEW TECHNIQUES

Most of the communication in social media is in textual form. While social media authors adhere to most rules of text production, the low level of editorial oversight, the perceived informality of the media, and the comparatively high degree of interactivity create a new communicative situation. There are no previous genres for this new type of communication — *new text* — to model itself after: new conventions for expression are created apace and we can expect several

new genres to appear eventually, blending features of established genres from paper-borne text with entirely new types of expression. This change may occur rapidly, in leaps and bounds, but firmly rooted in accepted textual practice as understood by the author — meaning that new texts do not necessarily transcend or break norms in every respect. [16, 20, 17, 18].

We claim that pre-compiled lexical resources — which work well for thoroughly edited, traditionally produced text — cannot be trusted to capture relations from new text. For example, in order to understand what someone means when they write on their blog that “my whip is the shiznit,” we need to know that “whip” in this context means “expensive automobile” and “the shiznit” means “good.” Consulting a standard dictionary will clearly not be of much help in this case; the only way to extract these relations is to mine the data itself.

We will in these experiments explore the challenges of practically acquiring new terminology, and of modelling similarity and relatedness of terms from observing realistic amounts of data rather than using manually compiled resources. Real-world deployment of terminology mining in social media requires handling the following issues:

Scalability: techniques intended to operate on social media must be applicable to very large data sets and have readiness to accommodate even more (and continuously increasing amounts of) data. It is not realistic to deploy methods which seize when the amount of data accrues over some habitable limit in an application area where volume really does matter.

Change: the techniques must be incrementally updateable and able to cope with constantly changing data. It is not realistic to deploy methods which require periodic re-compilation or re-organization to keep up with data evolution. Text analysis systems that are intended to work with social media and blogs must be able to handle semantic transformation, innovation, and permutation.

Noise: the techniques must be robust and not crumble under noisy and incomplete data. Any method that relies on non-trivial preprocessing (such as part-of-speech tagging, syntactic chunking, named entity recognition, language identification) or external resources (such as thesauri or ontologies) will be brittle in face of real-world data.

In this paper, we suggest using *Random Indexing* to handle these issues. In the following sections, we give an overview of

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CIKM'09, November 2–6, 2009, Hong Kong, China.

Copyright 2009 ACM 978-1-60558-512-3/09/11 ...\$10.00.

Random Indexing, and discuss how it can be used for coping with rapidly evolving word usage in social media. We apply the method to a sizeable collection of blogs, and provide a number of examples of how terminology mining in social media differs from using standard balanced corpora. In particular, we explore and investigate the following questions:

- Is the terminology really that different in social media and standard balanced corpora?
- Do we really need to mine social media in order to keep up with vocabulary variation?
- Do we really need large samples of data in order to capture word usage?
- Is the vocabulary growth different in social media as compared to standard balanced corpora?
- How do the semantic neighborhoods evolve when the size of the data increases, and how does the growth factor compare across different data genres?
- Do terms have different distributional patterns which characterize them usefully for terminology mining?

2. NEW TECHNIQUES REQUIRE RETHINKING OLD CONCEPTS

As argued in the previous section, social media constitute a semantically volatile domain, and if we intend to operate textually in such an environment we need to employ a methodology that can re-align its semantic model according to observed language use. A theoretical perspective that fits particularly well with this requirement is the view professed by structural linguistics that words can be characterized by the contexts in which they occur, and that semantic similarities between words can be quantified on the basis of distributional information. This idea, most concisely stated in Harris [3], has been enormously influential and has been operationalized by a family of statistical algorithms known as *word space models* [19, 13], which include well-known algorithms like Latent Semantic Analysis (LSA [8]) and Hyperspace Analogue to Language (HAL [10]).

These models collect distributional statistics in high-dimensional vectors called *context vectors*, which represent the distributional profile for terms (or whatever type of entity we are interested in). The context vectors contain co-occurrence counts that have been collected either by noting co-occurrence events with other words within a *context window*, as in the HAL-type of models, or by noting co-occurrence events within a document or other type of text region, as in the LSA-type of models. These two main types of models extract different types of similarities between terms. Sahlgrén [13] labels them paradigmatic and syntagmatic, since the HAL-type models will group together words that have been used with the same *other* words, while the LSA-type models will group together words that have been used together. A perhaps more enlightening characterization would be to label these different types of models *semantic* and *associative*. We will only discuss the former type of model in this paper, since we are primarily interested in semantic rather than associative similarity; knowing that “shiznit” means “good” is more useful than knowing it is associatively related to

“bro.”¹ Furthermore, LSA is not directly applicable to the current problem due to its well-known limitations regarding scalability and efficiency.

2.1 Semantic word spaces

Semantic (or HAL-type) word spaces produce context vectors by noting co-occurrence events within a context window that defines a region of context around each word. In the following example, the context window, indicated by “[],” spans 2 tokens on each side of the focus word “notion:”

...the seat [redefines the **notion** of sustainability] as a...

The context vector for “notion” in this example is 8-dimensional, since there are 8 word types, and it has non-zero values in the dimensions representing the words within the context window (“redefines,” “the,” “of” and “sustainability”).²

	the	seat	redef.	notion	of	sust.	as	a
notion	{1,	0,	1,	0,	1,	1,	0,	0}

The context window is then moved one step to the right, and the process repeated for the word “of,” then for “sustainability,” and then for “as,” etc. until the entire data has been processed. At the end of processing, each word type is represented by an accumulated n -dimensional context vector, where n is the size of the vocabulary of the data set, and each element records the number of times the word and the word represented by that dimension has co-occurred within the context window. The resulting matrix of context vectors is $n \times n$ and symmetric. It should be noted that in the original HAL model, co-occurrences are collected in only one direction within the context window, leading to a *directional* matrix in which rows and columns represent co-occurrence counts in different directions (i.e. with preceding and succeeding words).

If we now search through our n -dimensional vector space for the context vectors that are most similar to each other,³ we will find words that have been used in similar ways in our data and therefore have a semantic relationship, like synonyms and antonyms. An unsolved problem with these methods is how to distinguish words with the same meaning (i.e. synonyms) from words with the opposite meanings (i.e. antonyms). We will see several examples of this issue in the examples and experiments in Sections 3.2 and 4.2.

The most significant difference between various semantic word space models is the size and configuration of the context window within which co-occurrence counts are collected. The original HAL model uses a window spanning 10 words and collects co-occurrence counts asymmetrically within the window. Other models use small symmetric windows spanning two to three words on each side of the focus word [9, 13]. There have also been suggestions to utilize syntactic structure for configuring the context window. Padó

¹This is not to say associative word spaces cannot be of interest for text analysis applications — on the contrary, it is often of great interest to know which associations a certain term has (e.g. in buzz monitoring and brand name analysis).

²Note that we do not consider the word as co-occurring with itself.

³Similarity between context vectors are computed using any vector similarity measure, like Euclidean distance or, more commonly, because it normalizes for vector length, the cosine of the angles between the vectors: $\cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{|\vec{x}| |\vec{y}|}$

and Lapata [11] use dependency parsed data to produce semantic word spaces, but their results on a standardized synonym test (73% correct answers) is well below state-of-the-art results using standard context windows ($\approx 80\%$). Furthermore, syntactic analysis requires non-negligible amounts of preprocessing that would not be feasible in the present application.

2.2 Random Indexing

As shown in the previous section, context vectors produced from semantic word space models are n -dimensional, where n is the size of the vocabulary. This is not a viable approach when dealing with very large, and continuously evolving, vocabularies. An attractive solution to this problem is the Random Indexing framework [5], which allows for incremental accumulation of context vectors in a reduced-dimensional space whose dimensionality never increases.

This is accomplished by letting each word be represented by several randomly chosen dimensions instead of just one. For example, say that we have eight words in our vocabulary, as in the example above. Instead of using an 8-dimensional space where each word is represented by one dimension each, we can use a 4-dimensional space in which each word is represented by *two* dimensions, by using one negative value and one positive value as in the following example:

	d_1	d_2	d_3	d_4
the	{0,	-1,	+1,	0}
seat	{0,	0,	+1,	-1}
redefines	{-1,	+1,	0,	0}
notion	{+1,	0,	0,	-1}
of	{-1,	0,	+1,	0}
sustainability	{+1,	0,	-1,	0}
as	{-1,	0,	0,	+1}
a	{0,	0,	-1,	+1}

Such distributed representations are called *index vectors*, and they are high-dimensional (i.e. on the order of thousands) with a small number of +1s and -1s randomly distributed according to:

$$\vec{r}_i = \begin{cases} +1 & \text{with probability } \frac{\epsilon/2}{d} \\ 0 & \text{with probability } \frac{d-\epsilon}{d} \\ -1 & \text{with probability } \frac{\epsilon/2}{d} \end{cases}$$

where \vec{r}_i is the i th element of index vector \vec{r} , d is the pre-determined dimensionality, and ϵ is the number of non-zero elements (i.e. +1s and -1s) in the random index vectors.

These index vectors can be used to accumulate context vectors by the following simple algorithm: for each word in the data, update its zero-initialized context vector by adding the random index vectors of the words in the context window. For example, the context vector for “notion” in our example sequence “...the seat [redefines the **notion** of sustainability] as a...” is:

	d_1	d_2	d_3	d_4
notion	{-1,	0,	+1,	0}

which is the vector sum of the index vectors for “redefines,” “the,” “of,” and “sustainability.”⁴ This process is repeated

⁴Note that a context vector in our terminology is a *global* representation of *all* the contexts in which a word has occurred, and *not* a representation of a single occurrence.

every time we observe “notion” in our data, adding more information to its context vector. Note that the dimensionality of the context vector will remain constant, regardless of how much data we add. If we encounter a new word, we simply assign to it a random index vector, and update its zero-initialized context vector. Thus, no re-compilation is necessary when adding new data, making Random Indexing inherently incremental and scalable.

Random Indexing, and related methods like random mapping [6] and random projections [12], are based on the insight that choosing random directions in a high-dimensional space will approximate orthogonality, which means that by using the random index vectors to accumulate context vectors we will approximate a semantic word space using considerably less dimensions. Thus, if we collect the random indexing-accumulated context vectors for n words using d dimensions in a matrix $M_{n \times d}$, it will be an *approximation* of a standard words-by-words co-occurrence matrix $F_{n \times n}$ under the following condition:

$$M_{n \times d} \approx F_{n \times n} \times R_{n \times d}$$

where $R_{n \times d}$ is a matrix containing the random index vectors. Note that if the random vectors in matrix R are orthogonal, so that $R^T R = I$, then $M = F$. If the random vectors are nearly orthogonal, then $M \approx F$ in terms of the similarity of their rows (i.e. their context vectors).

The permutation operator

As Sahlgren et al. [14] demonstrate, Random Indexing also allows for incorporating word order information in the context vectors. This is achieved by permuting the random index vectors with regard to where in the context window the words occur. For example, in order to utilize word order information when accumulating the context vector for “notion” in our example sentence, we do:

$$(\Pi^{-2} \text{redefines}) + (\Pi^{-1} \text{the}) + (\Pi^1 \text{of}) + (\Pi^2 \text{sustainability})$$

where Π is a (random) permutation, Π^{-1} is its inverse, and the exponent n in Π^n signifies that the vector is permuted n times. This operation can be used either to incorporate word order by permuting index vectors by their position in the context window, or to use only the direction of the window (i.e. whether a word precedes or succeeds the focus word) as in the original HAL model, in which case we only use two permutations for preceding and succeeding words, respectively. Sahlgren et al. [14] show that *directional* context vectors outperform both standard unordered context windows and representations that take word order into account in a standardized synonym selection task.

In addition to producing state-of-the-art results in semantic tasks, the permutation operation also lets us retrieve the most frequent preceding and succeeding words. This is done by using the inverse permutation on the context vector for the word whose neighbors we want to examine, and then comparing this permuted context vectors to all *index* vectors. The words whose index vectors are most similar to this permuted context vector are the ones that tend to occur in the word’s immediate vicinity. Note that this allows us to use a semantic word space as a language model and predict the most likely preceding and succeeding words — what we will refer to as *directional similarities*.

Because of the state-of-the-art results in synonym detection tasks, and their ability to retrieve directional similarities, we use a small directional context window spanning two words to the left and two words to the right in the following experiments.

3. MINING IN WORD SPACES

Random Indexing is often claimed to be both scalable and efficient. In the remainder of this paper, we will put this claim to the test and apply Random Indexing to a sizeable collection of blog data. We will investigate the usefulness of the approach for terminology mining, and provide examples of both semantically and directionally similar terms to a number of target concepts that might be of interest for various kinds of social media analysis applications. In order to get an idea of how domain specific relations mined from social media are, we will compare these examples to ones produced using a medium-sized balanced corpus.

We will also look at the evolution and density of the semantic neighborhoods, and at how these properties compare between balanced corpora and blog data. One important reason to look at semantic evolution and density is to substantiate the assumption that “more is better” when it comes to capturing word usage. It also provides us with valuable insight into the nature of semantic word spaces, and their usefulness for terminology mining.

3.1 The Spinn3r data

The Spinn3r data set [2] is one of the currently largest publicly available data sets. It consists of some 44 million blog posts made between August 1st and October 1st, 2008. The data is arranged into tiers based on Spinn3r’s in-house blog ranking algorithm; tier 1 is the biggest sub-collection and contains the most “relevant” (as computed by the ranking algorithm) entries, tier 2 the second most relevant entries, and so forth. We use both tier 1 and tiers 2–13 in the following experiments.

The collection contains blog posts in many different languages; English being the most common, but Chinese, Italian, and Spanish are also frequent. We did not separate the different languages, since a multilingual environment is the natural habitat of social media, and any system intended to work with such data must be able to cope with multilinguality without the recourse to external resources. We will see several examples of where different languages collide in the semantic neighborhoods in the following experiments.

Before applying Random Indexing to the Spinn3r collection, we did a quick-and-dirty cleaning up of the data by removing non-alphabetic characters, downcasing alphabetic characters, and removing the Spinn3r xml tags. A substantial amount of noise remains after this naïve preprocessing, mostly in the form of html code, but also in the form of meta data from the Spinn3r xml format, which could have been removed by parsing the data more carefully. However, we wanted to simulate a noisy real-world environment in which careful preprocessing cannot always be afforded. We will see several symptoms of the noisy data in the examples in Section 4.2.

3.2 Nearest neighbor example

As an example of the kind of result that is typical when performing terminology mining using word spaces, we built a semantic word space from tiers 2–13 of the Spinn3r data,

good		bad	
great 0.91	✓	weird 0.86	✓
perfect 0.83	✓	sucky 0.86	✓
perfect 0.83	✓	scary 0.86	✓
pristine 0.81	✓	cool 0.85	≠
stable 0.80	✓	nasty 0.84	✓
grat 0.80	✓	dumb 0.84	✓
fantastic 0.80	✓	sad 0.84	✓
flawless 0.79	✓	lame 0.84	✓
mint 0.79	✓	creepy 0.84	✓
immaculate 0.79	✓	stupid 0.84	✓
geat 0.78	✓	dog 0.84	?
excellent 0.78	✓	shitty 0.83	✓
working 0.77	✓	quiet 0.83	?
decent 0.77	✓	romantic 0.83	?
excelent 0.77	✓	wierd 0.83	✓
ggod 0.77	✓	blind 0.83	?
nice 0.77	✓	prayer 0.83	×
rough 0.75	?	gun 0.82	×
goos 0.75	✓	lonely 0.82	✓
quiet 0.75	✓	boring 0.82	✓
excellant 0.75	✓	racist 0.82	✓
grate 0.75	✓	fuckin 0.82	✓
exellent 0.74	✓	cop 0.82	×
exelent 0.74	✓	fantastic 0.82	≠
execellent 0.74	✓	kid 0.82	×
noisy 0.74	?	dirty 0.82	✓
prestine 0.74	✓	selfish 0.82	✓
rare 0.73	✓	damn 0.82	✓
very 0.73	×	killer 0.82	?
restorable 0.73	?	mystery 0.82	?
superb 0.72	✓	broken 0.82	✓
positive 0.72	✓	young 0.82	?
strange 0.72	?	true 0.82	≠
cool 0.72	✓	boy 0.82	?
shallow 0.71	×	demo 0.81	?
pefect 0.71	✓	rough 0.81	✓
raffle 0.71	×	divorce 0.81	×
crowded 0.71	?	patch 0.81	?
phenomenal 0.71	✓	dream 0.81	×
excellect 0.71	✓	horrible 0.81	✓

Table 1: 40 nearest neighbors in semantic word space to “good” and “bad.” ✓ indicates viable (near) synonyms, × indicates errors, ? indicates uncertain cases, and ≠ indicates antonyms. The numbers are cosine similarities.

which constitutes some 600,000,000 word tokens.⁵ Table 1 shows the 40 nearest neighbors to “good” and “bad.” Neighbors that would count as (near) synonyms are indicated with (✓), errors are indicated with (×), strange — but still in some sense viable — neighbors with a question mark (?), and antonyms with (≠). Note that the presence of antonyms in the nearest neighbor lists is to be expected for two reasons: firstly, antonyms tend to occur in similar contexts; secondly, in some cases, semantic role reversal may flip the meaning of a term to resemble its antonym — a frequent occurrence for the word “bad.”

The most notable finding is that there *are* a number of domain-, genre-, or stylistic-specific terms among the nearest neighbors that would have been difficult, if not impossible, to foresee for a human analyst or by consulting a lexical resource. Examples include “pristine,” “immaculate,” and “mint” for “good” and “sucky,” “shitty,” and “creepy” for “bad.” Another very typical effect when analyzing social

⁵Using 2,000-dimensional vectors and a directional context window consisting of the four nearest surrounding words.

media is that misspellings often show up among the nearest neighbors: “prefect,” “grat,” “geat,” “excelent,” “ggod,” “goos” and so on.

The words marked with a question mark would most likely be ranked as unrelated if the nearest neighbor list would be evaluated with a standard lexical resource like a thesaurus or a dictionary. However, it is possible that words like “rough,” “noisy,” “strange,” and “crowded” actually have very positive loading in the Spinn3r data, and that they therefore quite correctly should be related with “good.” Similarly, “dog,” “quiet,” and “romantic,” are possibly very negative terms in the blog data, and therefore correctly related to “bad.”

Although not visible in the examples in Table 1, we do encounter multilingual effects in the nearest neighbor analysis of the Spinn3r data, which is to be expected since we did not attempt language separation. For example, extracting the nearest neighbors for “love” we find words like “Norge” (i.e. “Norway” in Norwegian), “Stavanger” and “Tromsö” (both Norwegian cities), which can be explained by the fact that “Love” is a Norwegian male surname. This also accounts for the fact that a number of other person names show up in the nearest neighbor list for “love” — e.g. “hshm,” “svevo,” “nosanto,” “drew,” “nomi,” and “takako.”

4. SCALING UP

In the remainder of this paper, we use data from tier 1 of the Spinn3r collection, which after our quick-and-dirty cleaning-up consists of some 1,000,000,000 word tokens. We constructed a semantic word space using 2,000-dimensional vectors and a directional context window spanning four surrounding words, which took around a day to compute using brute force (i.e. without any optimization such as caching). As comparison, we also include examples computed using the same parameter setting for the British National Corpus (BNC) — a balanced English corpus containing approximately 100,000,000 words.

4.1 Evaluation

Evaluation of automatically extracted lexical resources from non-standard data is notoriously difficult, since there typically does not exist a gold standard available for comparison and benchmarking. This is particularly true for terminology mining in social media, where — as we have argued — the dynamic nature of language use prohibits the compilation of lexical resources, and makes standard benchmarking procedures inapt. Stretching this line of reasoning, we could say that whatever we find in the data must be the truth for those particular data. However, that would be begging the question; arguably, systematic benchmarking and evaluation are crucial to useful natural language engineering and application design.

We believe that evaluation of learned resources must be built on an experimental process based on hypotheses informed by some understanding of textual reality, rather than computational expediency, and that results must also be evaluated by the qualities of the representation per se, and not only by their application to some noisy and imprecise task. Using task-based evaluations, while guaranteeing a measure of validity for the experiment, risks swamping the effects of the representation in a context where other factors may induce variation, not obviously visible to the experimenter.

In this study, we will focus on questions preceding those of general applicability and usefulness. In particular, we are interested in factors that *affect* the semantic representations, such as the diversity of language use and the evolution of the semantic neighborhoods. Our main motivation for performing this study is that intrinsic evaluation of resources such as word spaces needs to be formalized to model variation across collections and over time. This study is a step towards establishing measures for understanding the intrinsic variation of such representations.

4.2 Target concepts

As discussed in the previous section, our goal here is not to evaluate how well semantic word spaces gathered from blog data replicate a thesaurus or a dictionary (since that is irrelevant from an application-driven perspective). Rather, our ambition in this paper is to investigate and characterize the properties of blog-induced word spaces as compared to standard corpus-built ones. However, since one of the most obvious applications of semantic analysis of social media is buzz monitoring applications, we focus here on a number of hand-chosen target concepts that are relevant for, and often used by, such systems:

Buzz monitoring: *love, hate, recommend*

Open source intelligence: *attack, bomb, terrorism*

Epidemiology: *flu, infection, symptom*

Climate: *ecology, ecosystem, environment*

Tables 2 and 3 show the five most semantically and the four most directionally related terms for each target concept. For example, the left column for “love” contains its two most commonly preceding terms, whereas the right column contains its two most commonly succeeding terms. The column below “love” shows its five nearest semantic neighbors.

Nearest neighbor lists are admittedly less clarifying, difficult to interpret, and may even be potentially delusive since authors often have the unfortunate tendency to weed out unfavorable examples. We include these un-edited lists here in order to show the varying quality of the results; some words, like “infection” and “bomb,” have very relevant nearest neighbors in both tables, while other words, like “ecosystem” and “ecology,” have not.

Also, we want to show the differences between edited corpora and noisy data: the nearest neighbors produced from the Spinn3r data are clearly more noisy (see e.g. “terrorism”), and feature numerous spelling variations and surface noise introduced by the simplistic preprocessing (see e.g. “recommend” and “infection”). Furthermore, we include the nearest neighbor lists because we focus on a small number of target concepts, simulating a real-world application scenario where a human analyst consults such lists in order to extract associations and keep up with vocabulary variation and development.

The reason we only include four directional neighbors in the examples is that directional analysis using the permutation operator in Random Indexing can capture frequently occurring constructions — like “car bomb” — but less frequent co-occurrences will drown in the noise inherent in the Random Indexing algorithm. This means that looking further down the directional neighbor lists than, say, the two highest correlated words will result in an increasing rate of

←	love	→	←	hate	→	←	recommend	→
fall	asleep	mystique	love	loathe	because	also	criticize	kilodalton
god	grace	affair	really	despise	man	report	conclude	joe
.	foul	.	.	love	.	.	criticise	.
.	hate	.	.	adore	.	.	propose	.
.	enjoy	.	.	enjoy	.	.	suggest	.
←	infection	→	←	symptom	→	←	flu	→
hiv	colonisation	profā	physical	problem	may	cold	influenza	virus
pylori	aids	amore	gastrointestinal	event	sign	bout	vaccinia	epidemic
.	pylori	.	.	defect	.	.	herpes	.
.	positive	.	.	characteristic	.	.	polio	.
.	oxbury	.	.	feature	.	.	aids	.
←	attack	→	←	terrorism	→	←	bomb	→
heart	thudding	over-used	against	door-frame	act	ira	firebomb	attack
under	pounding	thereunder	prevention	backdrop	weevil	car	car-bomb	explode
.	thumping	.	.	door-jamb	.	.	terrorist	.
.	thud	.	.	d-mark	.	.	landmine	.
.	thump	.	.	deutschmark	.	.	unprovoked	.
←	ecology	→	←	environment	→	←	ecosystem	→
terrestrial	wedgewood	usage	computing	system	minister	marine	invertebrate	type
misc	none	tag	distributed	strategy	secretary	natural	vegetation	cookery
.	misc	.	.	defence	.	.	selection	.
.	compute	.	.	cabinet	.	.	habitat	.
.	advert	.	.	interior	.	.	phenomenon	.

Table 2: Semantic and directional nearest neighbors extracted from the BNC. The first column under ← for each target shows the 2 most likely preceding words, the middle row shows the 5 nearest semantic neighbors, and the last column under → shows the 2 most likely succeeding words.

random neighbors.⁶ Another, somewhat more informed approach to extract relevant directional neighbors would be to use a threshold for the cosine similarities. As an example, the average cosine similarity for our targets to their nearest directional neighbor is ≈ 0.49 , to the second nearest directional neighbor ≈ 0.26 , and to the third nearest directional neighbor ≈ 0.24 . This can be contrasted with 0.85, which is the average cosine similarity to the *ten* nearest semantic neighbors, indicating the difference in reliability between these measures.

There are a number of findings of practical import in these lists. First of all, we note that antonyms turn up among the nearest neighbors in semantic spaces built from both types of data — see e.g. “love” and “hate.” This artefact of semantic spaces has already been discussed in Section 3.2 above. We also find that certain words have the same nearest neighbors in both types of data — e.g. “hate” has “despise” and “loathe” as nearest neighbors in both the BNC and Spinn3r spaces. Furthermore, the most likely preceding words for “attack” (i.e. “heart” and “under”) are the same in both spaces, as are a number of strong collocations, like “hiv infection,” “car bomb,” and “bomb attack.”

This indicates the stability of semantic neighborhoods across domains and data; theoretically, we would expect semantic neighborhoods to remain relatively stable across different styles and genres compared to associations, which are by nature domain specific. Semantic relations, on the other hand, are constraints on vocabulary choice and are therefore less likely to be subject to individual variation. However — which is the point in this paper, and as these examples demonstrate — there are also domain specific synonyms that are only used in a certain language sample, and that would not be foreseeable by a human analyst. Such domain spe-

cific nearest neighbors can be, e.g., the collocation “bird flu” and “algeria” as a nearest neighbor for “bomb.”

The fact that we do find stable semantic neighbors across the different word spaces suggests that we might be able to use the overlap between several domain-specific word spaces as a generic semantic representation. A simple example of this idea is demonstrated in Table 4, which contains those terms that occur in both the BNC and Spinn3r word spaces among the 100 nearest semantic neighbors to four of the targets. Obviously, these semantic neighbors are less impressive when it comes to precision than when it comes to recall, but they do describe the semantic domains of the target words quite well. If we only use the 5 nearest semantic neighbors in both spaces, we get the terms displayed below, showing a much higher semantic precision and again demonstrating the conflation of antonyms and synonyms in word space:

hate: *detest, adore*

love: *adore, hate, loathe*

attack: *ban*

environment: *theatre*

Counting the overlap between the 100 nearest semantic neighbors in the BNC and Spinn3r word spaces for our targets gives results ranging from 0 and 1 overlap (for “symptom” and “ecosystem”) to 21 and 22 terms in common (for “bomb” and “attack”). This rather low level of overlap might seem surprising in view of the general character of the chosen target concepts in this study: one might expect the general properties of the English language to provide considerable higher rate of overlap across domains. This is a further indication of the necessity of domain-specific terminological mining.

As a last reflection on the examples from the Spinn3r data in Table 3, we can note that there seems to be a qualitative difference between the left and right directional neighbors: the latter contain more noise, particularly in the Spinn3r

⁶It should be noted that in some cases there are actually viable terms further down the directional lists; e.g. “yeast” and “sinus” are the third and fourth nearest left directional neighbors for “infection,” and “roadside” and “nuclear” are the third and fifth left directional neighbors for “bomb.”

←	love	→	←	hate	→	←	recommend	→
much	hate	him	love	despise	him	highly	reccommend	marbig
id (I'd)	lovebr	allentowns	really	love	them	papercut	reccommend	allbrittons
.	loathe	.	.	loathe	.	.	reccommended	.
.	despise	.	.	dislike	.	.	regarded	.
.	adore	.	.	hatebr	.	.	recommended	.
←	infection	→	←	symptom	→	←	flu	→
hiv	infections	peculiar	another	person	free	bird	flubr	linzeeluvcheez
ear	infectionbr	bumiputeras	only	problem	consolidated	avian	feeders	prosser
.	infectionsbr	.	.	drawback	.	.	bcfmo	.
.	integrase	.	.	downside	.	.	mothrad	.
.	infectionfont	.	.	part	.	.	plasticities	.
←	attack	→	←	terrorism	→	←	bomb	→
heart	attackbr	against	international	wota	extremism	atomic	bombs	attack
under	palpitations	alcyone	fundamentalism	fundamentalism	prosaico	car	bombing	iran
.	rending	.	.	otcqx	.	.	blasts	.
.	taenor	.	.	karting	.	.	algeria	.
.	throb	.	.	baccalaureate	.	.	exploded	.
←	ecology	→	←	environment	→	←	ecosystem	→
media	cognition	vol	info	strategy	newsfeed	inkoper	gwyn	probably
knowledge	sociology	social	ecoearth	plan	vast	vlaming	ypl	cushmans
.	stubbleblog	.	.	curriculum	.	.	butwait	.
.	brandtobe-	.	.	atmosphere	.	.	floccinaucinihi-	.
.	determined	.	.	efficient	.	.	lipilification	.
.	biology	longsword	.

Table 3: Semantic and directional nearest neighbors extracted from the Spinn3r tier-1 data. The first column under ← for each target shows the 2 most likely preceding words, the middle row shows the 5 nearest semantic neighbors, and the last column under → shows the 2 most likely succeeding words.

love	ecology	terrorism	environment
hate	art	witchcraft	theatre
loathe	architecture	socialism	garden
adore	hypocrisy	racism	club
despise	ethics	colonialism	media
detest	chaos	diplomacy	industry
miss	music	nationalism	renaissance
dislike	politics	paranoia	market
swear	privacy	sexuality	economy
guess		slavery	hill
feel		motherhood	landscape
		alcohol	clan

Table 4: Terms that occur among the 100 nearest semantic neighbors to four targets in both the BNC and Spinn3r word spaces.

examples. The fact that the right context presents greater variation than the left context is not by itself surprising, since the left context is governed by phrase-level syntactic constraints, particularly in fixed-word order languages such as English, and especially for nouns. A symptom of this is the higher average cosine similarity to the nearest left directional neighbor (≈ 0.59) than to the nearest right directional neighbor (≈ 0.39). This effect is more pronounced for the Spinn3r examples: “allentowns,” “bumeputeras,” “prosaico,” and the right directional neighbors for “recommend,” “flu,” and “ecosystem” are all nonsensical. This is an indication of how the lesser level of editorial control in social media affords the authors greater freedom in lexical choice; the overriding constraints from the English language are not relaxed to the same extent as the stylistic and topical ones might be.

4.3 Semantic evolution

It may seem natural to assume that, since word spaces are statistical algorithms, adding more data will improve the quality of the word spaces. Since we evaded the question

of how to assess the quality of semantic resources in this paper, we can rephrase the question in the following terms: how much does the semantic neighborhoods evolve when we add more data?

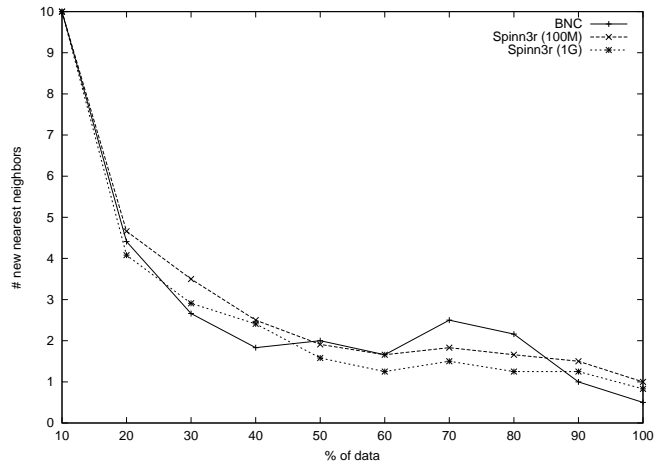


Figure 1: Semantic evolution in word space for the target concepts. The y-axis indicates the number of new terms among the ten nearest neighbors. The x-axis shows percentage of total data used.

Figure 1 shows how the semantic neighborhoods evolve when adding more data for three different data sets — the BNC, the first 100 million words from the Spinn3r data, and the full one billion word Spinn3r tier-1 data. The y-axis shows the number of new terms among the ten nearest neighbors for our 12 target concepts, and the x-axis shows percentage of total data used. As an example, the ten nearest neighbors to “love” when having seen 90% of the Spinn3r data is:

hate hope loathe loooove adore loooooove looove loved detest miss

and when having seen 100% of the data:

hate hope loathe loooove adore looooooooooove looove loooooove miss detest

which differs only by one word (disregarding the order): in the latter list is “loooooooooove” instead of “loved.”

The tendency is the same in all word spaces; adding more data alters the neighborhoods slightly, but the evolution of the word spaces stagnates quite quickly. The evolutionary process seems to follow a power-law-related distribution, describing the progressive semantic saturation of the local neighborhood of the focus term, and after we have doubled the amount of data a couple of times we merely see on average one new neighbor in the semantic neighborhoods when adding more data. The neighborhood for some words change more than others. For example, “ecosystem,” “infection,” and “flu” tend to replace more neighbors when adding data than “love,” “hate,” and “bomb.” We suspect this is due to the former words’ relative semantic promiscuity — words that have a broader usage will also modify its semantic neighborhood more frequently than words with a relatively static usage.

It seems that adding more of the same type of data does not alter the semantic neighborhoods that much after a certain point. Again, this is related to the relative stability of semantic neighborhoods discussed in the previous section. However, adding more of *another type* of data would certainly have a discernible effect on the semantic neighborhoods; if data discussing swine flu would be added to a word space in which flu is related to bird, we would certainly see an evolution in the semantic neighborhood of “flu.” The question is whether the evolution of semantic neighborhoods would follow the same type of power-law-like distribution even when continuously adding diverse data? And whether there is any way to determine how semantically homogeneous a data set is, so as to thereby predict the evolutionary rate of semantic neighborhoods? We leave the former of these questions open for future research, and suggest one direction to approach the latter question in the following section.

4.4 Semantic density

One way to arrive at an indication of the semantic homogeneity of a data set is to compute the density of neighborhoods in word space, as suggested by [15]. The idea is that very homogeneous data have very dense semantic neighborhoods, since words occur in very uniform contexts. Thus, if we extract the n nearest neighbors to a target word, and then the n nearest neighbors to each of the target word’s nearest neighbors, we can quantify this neighborhood’s density as the density measure d_n by simply counting the total number of unique words thus extracted. The maximum number of unique words is $n \times n$, indicating an extremely dispersed neighborhood; the minimum number of unique words is n , indicating an extremely homogeneous word usage, where all neighbors form an interconnected set.

Figure 2 shows the density of the semantic neighborhoods of our target concepts for d_{10} over ten different sizes of the data sets (i.e. the BNC, the first 100 million words from the Spinn3r data, and the full one billion word Spinn3r tier-1 data). The y -axis shows the average density measure for our 12 target concepts, and the x -axis shows percentage of total data used. As can be seen in the figure, the densities

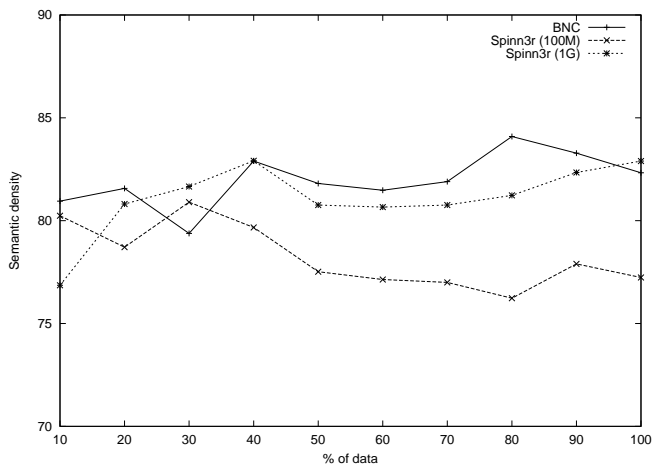


Figure 2: Semantic density of neighborhoods in word space. The y -axis shows d_{10} , indicating the number of unique terms among the 10 nearest neighbors of the 10 nearest neighbors to each target concept, with scores ranging from 10 to 100. The x -axis shows percentage of total data used.

of the semantic neighborhoods for our target concepts revolve around 80, and are comparable in the different spaces. Perhaps not very surprisingly, the density for the balanced BNC data seems to be somewhat higher than in the collection of blog data (around 82 for the BNC and around 77 for the Spinn3r data).

This speaks of the more focused topical character of the BNC data. Where most of the BNC is constituted of printed text, the Spinn3r data treat large numbers of topics ranging from the whimsical and personal to the factual and informative, none primarily intended for printed publication. Table 5 gives an indication of this. Two classic measures of vocabulary richness are given for the two test collections [21, 4]. Their diverging results show that the Spinn3r data present more singleton term occurrences than the BNC data; they also show that the variability of the vocabulary in general is greater for the BNC. This observation demonstrates the effect of editorial processing on text evident in the BNC data: less misspellings and one-off terms; more variability in the greater pattern of language use.

	Spinn3r	BNC	LA Times
Honoré’s R	4448	4148	3831
Yule’s K	53.6	69.8	106.22

Table 5: Two measures of vocabulary richness, computed on the two test collections. Honoré’s R, if large, indicates many hapax legomena, single occurrences of terms. Yule’s K, if large, indicates a variable vocabulary. The data for one year of LA Times are given for reference purposes.

5. DISTRIBUTIONAL PROMISCUITY

The form of distributional analysis studied in these experiments — based on pointwise occurrence and co-occurrence statistics — deliver better results for some terms than for others. As we have seen in the examples throughout this

paper, some words have very relevant nearest semantic and directional neighbors in word space, while others have more or less random neighbors. Since the differences in comprehensibility between these terms for the human language user is small, this suggests that there may be further characteristics in term usage available for modelling in a learning framework. Previous studies have extended the study of pointwise statistics of term occurrences to *burstiness* or repeat occurrences of terms [7, 1], using Katz’ Poisson mixture models which appear to capture more of the topicality of terms than the pointwise frequency estimates are able to. However, the model we utilize here makes it possible to study the co-occurrence neighborhood itself, to examine and compare the diversity of the neighborhoods between terms. Our hypothesis is that terms which are *distributionally promiscuous* will have a more diffuse global representation in a co-occurrence-based scheme such as ours.

This question merits further systematic study, but given the varying utility of the target terms given in the previous section, we can study their respective qualities within our representation. From a practical standpoint, the question is whether we can determine the distributional suitability of a term apriori, without wasting much computational effort to model its neighborhood? Can this be done from first observable characteristics of the term? Can we somehow determine this by merely inspecting the intrinsic properties of the representation? Table 6 gives some candidate measures. We use — as a simplified version of the heuristic for parameter estimation given by Sarkar et al [1] — the ratio between Katz’ α^7 , and Katz’ γ^8 , estimated from separate newspaper data; we use the d_{10} density measure from Section 4.4 [15]; and we use the standard deviation σ of the context vectors in the Random Indexing word space.

Of these three candidate measures, the ratio between Katz’ two measures gives low scores to *ecosystem*, *terrorism*, and *ecology*, with fairly random neighbors as given in Tables 2 and 3 and high scores to *love*, *attack*, and *recommend*, all three usefully modelled terms. Similarly, the standard deviation gives a strong indication of predictive power, with high σ tending to be more characteristic of better terms, indicating that words with a fairly tightly held context are better targets for this technology. The density measure yields somewhat more equivocal results. These candidate measure serve here to indicate that the distributional character of terms can be analyzed to show correlation with their usefulness for distributional semantic modelling; they need considerable more refinement for true predictive power.

6. CONCLUSIONS

The point of this paper is to critically discuss the task of terminology mining in social media, and in particular to explore the viability of using scalable statistical approaches such as Random Indexing for this task. Our explorations have shown both possibilities and limitations of the proposed approach, and we have identified a number of interesting properties and directions for further study and application.

In general, we believe that the formal study of the characteristics of high-dimensional representations of linguistic

⁷The estimated probability of occurrence of the term in a document, (essentially the collection frequency of a document, df) [7].

⁸The estimated probability of a repeat occurrence given a first observation within the given text [7].

	Katz’ α/γ	d_{10}	σ
love	0.23889	41	16390
hate	0.08538	26	3308
recommend	0.14455	38	1059
infection	0.01546	76	230.2
symptom	0.03314	35	97.09
flu	0.01764	81	406.4
attack	0.17663	63	1792
terrorism	0.00906	67	947.0
bomb	0.03041	56	583.1
ecology	0.01636	47	81.70
environment	0.18243	45	2617
ecosystem	0.00659	55	187.4

Table 6: Individual distributional and intrinsic characteristics of the target concept terms.

information has been neglected in favor of the study of their applicability to general information access tasks, such as search engine index implementations, or semantic similarity extraction. The research field in general still has rather vague notions of how the make-up of the representation alters the properties of the semantic model; this study points to a number of representational issues, some of which may be artefacts of the high-dimensional representation.

On the other hand, this study also shows the practicability and robustness of scalable approaches such as Random Indexing, across collections, and even for *new text* such as found in user-generated media, which, compared to edited media sources, is characterized by:

- noise,
- multilinguality and language mixtures,
- variation, both individual and genre-based,
- domain specificity, with a large variety of domains and communicative communities.

Several of our observations support these general contentions. It is also clear that the variation given in user-generated media takes the form of very productive exploration of any given synonym space. The multitude of synonyms for the relatively basic notion of “good” is a case in point. Where established and experienced writers might work with structural and compositional features of the text, less experienced writers are prone to expending their creative energy on synonymy.

In summary, our main findings are:

- Dynamic terminology mining, rather than set lexical resources, is likely to be crucial to any task requiring high recall of materials from social media, and to capture new coinage and shifting usage of words in the language.
- More is not necessarily better after a certain point; the evolution of semantic neighborhoods appears to asymptote towards a fairly stable state. However, since even a small change in a semantic neighborhood has the potential of completely changing the effectiveness of a lexically based filter or search pattern, some model and prediction of the relative rate of change will be necessary.

- Semantic neighborhoods seem to be relatively stable across different data sources and sizes of text collections. This observation is contrary to most expectations in previously published work and will have ramifications for the further study and implementation of the intrinsic dimensionality of high-dimensional semantic models.
- Word space models will need to develop methods to distinguish antonyms from synonyms in order to be of practical use as terminology mining tools for human analysts.
- There is a need for methods that allow us to characterize and typologize terms with respect to their distributional properties. Such methods would allow us to predict the usefulness and applicability of terms for statistic terminology mining.

7. REFERENCES

- [1] A. D. R. Avik Sarkar, Paul H Garthwaite. A bayesian mixture model for term re-occurrence and burstiness. In *Proceedings of the 9th Conference on Computational Natural Language Learning (CoNLL)*, pages 48–55. Association for Computational Linguistics, Ann Arbor, June 2005.
- [2] K. Burton, A. Java, and I. Soboroff. The ICWSM 2009 spinn3r dataset. In *Proceedings of the Third Annual Conference on Weblogs and Social Media (ICWSM 2009)*, San Jose, CA, May 2009.
- [3] Z. Harris. *Mathematical structures of language*. Interscience Publishers, 1968.
- [4] A. Honore. Some simple measures of richness of vocabulary. *Association for Literary and Linguistic Computing Bulletin*, 7:172–177, 1979.
- [5] P. Kanerva, J. Kristofersson, and A. Holst. Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society, CogSci'00*, page 1036. Erlbaum, 2000.
- [6] S. Kaski. Dimensionality reduction by random mapping: Fast similarity computation for clustering. In *Proceedings of the International Joint Conference on Neural Networks, IJCNN'98*, pages 413–418. IEEE Service Center, 1999.
- [7] S. Katz. Distribution of content words and phrases in text and language modelling. *Natural Language Engineering*, 2(1):15–60, 1996.
- [8] T. Landauer and S. Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104(2):211–240, 1997.
- [9] J. Levy, J. Bullinaria, and M. Patel. Explorations in the derivation of word co-occurrence statistics. *South Pacific Journal of Psychology*, 10(1):99–111, 1998.
- [10] K. Lund, C. Burgess, and R. Atchley. Semantic and associative priming in high-dimensional semantic space. In *Proceedings of the 17th Annual Conference of the Cognitive Science Society, CogSci'95*, pages 660–665. Erlbaum, 1995.
- [11] S. Padó and M. Lapata. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199, 2007.
- [12] C. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala. Latent semantic indexing: A probabilistic analysis. In *Proceedings of the 17th ACM Symposium on the Principles of Database Systems*, pages 159–168. ACM Press, 1998.
- [13] M. Sahlgren. *The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. PhD Dissertation, Department of Linguistics, Stockholm University, 2006.
- [14] M. Sahlgren, A. Holst, and P. Kanerva. Permutations as a means to encode order in word space. In *Proceedings of the 30th Annual Conference of the Cognitive Science Society, CogSci'08*, pages 1300–1305, Washington D.C., USA, 2008.
- [15] M. Sahlgren and J. Karlgren. Counting lumps in word space: Density as a measure of corpus homogeneity. In *Proceedings of the 12th Symposium on String Processing and Information Retrieval, SPIRE'05*, 2005.
- [16] M. Santini. Interpreting genre evolution on the web: Preliminary results. In J. Karlgren, editor, *New Text. Proceedings from the workshop on New Text: Wikis and blogs and other dynamic text sources, held in conjunction with EACL*. ACM, Trento, Italy, 2006.
- [17] M. Santini. Characterizing genres of web pages: Genre hybridism and individualization. In *HICSS '07: Proceedings of the 40th Annual Hawaii International Conference on System Sciences*, Washington, DC, USA, 2007. IEEE Computer Society.
- [18] M. Santini, A. Mehler, and S. Sharoff. Riding the rough waves of genre on the web: Concepts and research questions. In A. Mehler, S. Sharoff, and M. Santini, editors, *Genres on the Web: Computational Models and Empirical Studies*. Springer, Berlin/New York, 2009.
- [19] H. Schütze. Word space. In *Proceedings of the 1993 Conference on Advances in Neural Information Processing Systems, NIPS'93*, pages 895–902, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.
- [20] M. Tavosanis. Linguistic features of italian blogs: literary language. In J. Karlgren, editor, *Proceedings of the EACL workshop on New Text: Wikis and blogs and other dynamic text sources*. European Association of Computational Linguistics, Trento, Italy, 2006.
- [21] G. U. Yule. *The statistical study of literary vocabulary*. Cambridge University Press, Cambridge, 1944.