A Graph Based Approach to Text Normalization

Abstract

Keywords: Text Normalization, Twitter, microblogs, social media

1. Introduction

Social text has become an enormous part of our lives. We are moving towards to an era that we will be talking using machines more than we talk to each other. Social platforms make mass amounts of people communicate via typed or transcribed text. That is the era that the news are spreading digitally via social media other than edited newspaper articles.

This has been also starting of a new era for text analytics researches. The recent studies on social media such that Stock Prediction[1], politeness detection[2], disaster detection[3] tries to lighten up the road of this digitized future of ours.

Therefore analyzing social media text is a challenge for itself. Due to its noisy nature, many NLP tools are performing poorly on social media text[4]. The problems that noise in the social media text generates for NLP tools can be overcome by some preprocessing steps.

Unlike spoken and written language, digitized language has its own form and nature. Since the beginning of World Wide Web, internet has it's own slang. *lol* meaning *laughing out loudly*, *xoxo* meaning *kissing*, *4u* meaning *for you* are the oldest examples of this slang. Everyday new slangs as well as new words such as iTunes and new abbreviations are coming up. It is a huge, an evolving language that has long gone behind the reach and control of spellcheckers and slang dictionaries.

ppl	people
havent	haven't
tmr	tomorrow
soooo	so
sooon	soon
raight	right
raight	alright

r	are
mor	more
doin	doing
n	and
friiied	fried
finge	finger
kissin	kissing

Table 1: Example of noisy tokens and their normalized form

Text normalization is a preprocessing step to restore noisy forms of text to its original(canonical) form[5] to make use of NLP applications or more broadly to understand the digitized text better. For example *talk 2 u later* can be normalized as *talk to you later* or similarly *enormoooos, enrmss, enourmos* cand be normalized as *enormous.* Those noisy tokens are referred as Out of Vocabulary(OOV) words. Normalization task is restoring OOV words to their In Vocabulary(IV) form.

However not every OOV word should be considered for normalization. The social text is continously evolving with new words and named entities that are not in the vocabularies of the systems [6]. The OOV tokens that should be considered for normalization is referred as ill-formed words. Oppositely an OOV word can sometimes lexically fit an IV word (ex: *tanks* is both an IV word and OOV word with the canonical form *thanks*). The task of recognizing which tokens are OOV, and which of those are ill-formed are beyond the scope of this paper.

Unlike the clean text, noisy text is difficult to model using standard language models. Due to noisy nature of the OOV tokens versions of a text includes multiple different OOV tokens (ex: Table 2). It is even more difficult to reach the correct version of the phrase when there is more than one OOV word in the text. A high performance normalization system should been capable of following the information a noisy token includes as good as in modeling well formed tokens.

with a beautiful smile
with a **beatiful** smile
w a beautiful smile
wit a beautiful **smil**
wth a **btfl sml**
w a **btfl smle**

Table 2: Example noisy n-grams

In this paper we propose a new approach to text normalization. A graph based model which can benefit from both lexical, contextual and grammatical features of social text.

2. Method

In this paper, we propose a graph based approach that models both contextual similarity features and lexical similarity features among an OOV word to be a normalized and the candidate IV words. A high level overview of our system is shown in Figure 1. An input text is first preprocessed by tokenizing and Part-Of-Speech (POS) tagging. If the text contains an OOV word, the normalization candidates are chosen by making use of the contextual features which are extracted from a pre-generated word-relatedness graph, as well as lexical similarity features. Lexical similarity features are based on edit distance, longest common subsequence ratio, and double metaphone distance. In addition, a slang dictionary is used as an external resource to enrich the normalization candidate set. The details of the approach are explained in the following sub-sections.

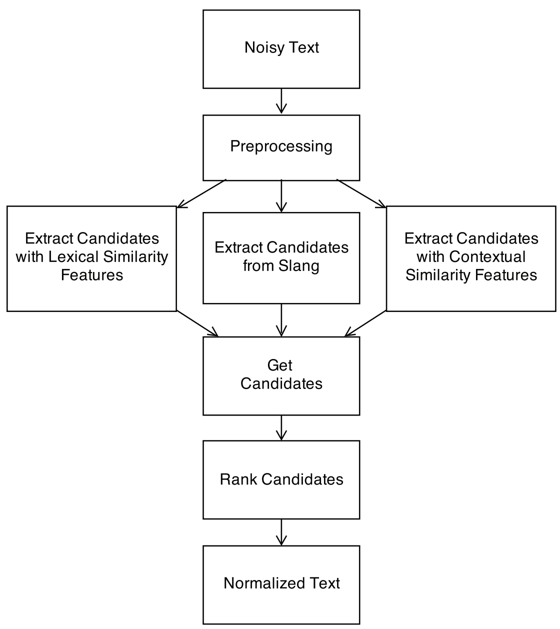

Figure 1: High level overview of our system

2.1. Preprocessing

Tokenization is the first step in our system. Tokenization is the process of breaking the text into words, numbers, symbols, emoticons or in other words the smallest meaningful elements within/of the text called tokens. After tokenization, next in the pipeline is POS tagging each token using a social media pos tagger. Unlike the normal pos taggers social media pos taggers [7][8] provide a broader set of tags that is special to the social text. By this extended set of tags we can identify tokens such as discourse markers (rt for retweets, cont. for a tweet whose content follows up in the coming tweet) or urls and we can process those tokens within their context.

As in Table 3, after preprocessing, each token has a POS tag with a confidence measure. Later, we make use of these confidence scores in calculating the weight of edges in our context graph.

Token	POS tag	Accuracy
with	P	0.9963
a	D	0.998
beautiful	A	0.9971
smile	G	0.9712

Token	POS tag	Accuracy
w	P	0.7486
a	D	0.9920
beatiful	A	0.9733
smile	N	0.9806

Table 3: POS tagger output of samples

2.2. Graph construction

The graph is build using a big dataset of social media text. After preprocessing, we traverse each entry in the dataset and extract nodes and edges.

We define a node with four properties *id, oov, freq, tag*. The token itself plus it's POS tag forms the *id* field. *freq* property indicates the node's frequency count in the dataset. *oov* field is set to True if the token is a OOV word. Following Han et al. we used GNU Aspell dictionary (v0.60.6) to determine whether a word is OOV or not.

In the word-relatedness graph, each node is a unique set of a token and a POS tag (see Table 4). This helps us to identify the tokens not only by lexically and contextually but also (in terms of POS tags) gramatically.

Let's$_L$ start$_V$ this$_D$ morning$_N$ w$_P$ a$_D$ beatiful$_A$ smile$_N$.$_,$

Tokens	Let's, start, this, morning, w, a, beatiful, smile, .
Nodes	Let's|L, start|V, this|D, morning|N, w|P, a|D, beatiful|A, smile|V, .|,
Edges	{Let's|L, start|V , distance:1},{Let's|L, this|D, distance:2}, ... {a|D, beatiful|A, distance:1}, {a|D, smile|V, distance:2}, {beatiful|A, smile|V, distance:1}

Table 4: Sample sentence with POS tags, Tokens, The Word-relatedness Graph Nodes and Edges

For example if the token *smile* has been seen frequently as a Noun and a Verb and not in other forms in the dataset (Ex: Table 5), this means that it is not a good candidate for a Pronoun OOV token(that is a Pronoun). On conversely, if lexically and contextually similar enough, *smile* is(can be) a good candidate for a Noun or Verb OOV token.

Edges are built upon(depending on) the relatedness metrics defined. For

node id : smile|A	freq : 3	oov : False	tag : A
node id : smile|N	freq : 3403	oov : False	tag : N
node id : smile|V	freq : 2796	oov : False	tag : V

Table 5: Example nodes including token smile with a frequency greater than 0

two nodes to be classified as related they have to satisfy both two rules:

- Tokens in an entry are conceptually related if they co-occurs within a word distance of d_t.
- Each node of an edge should have a minimum frequency of f_t in the whole dataset.

The edges follow the flow in the entries thus have a direction from the earlier seen token to the coming token. For example the edges in Table 6 would be derived from a text including the phrase "with a beautiful smile". The *from* property indicates the first word and *to* is the latter in the phrase. Direction and the distance together hold a unique triplet. For each two node with a specific distance there is an edge with a positive weight if that two nodes are related. Each co-occurrence of two related nodes increases the weight of the representing edge with an average of nodes' POS tag confidence score in that specific entry. If we are to expand the graph with our example phrase using the given POS tags and accuracies from Table 3, the increase in the weights would be respectively $0.9963 + 0.9712/2$, $0.998 + 0.9712/2$ and $0.9971 + 0.9712/2$.

from : with|P, to : smile|N, dis : 3, weight : 10.47095
from : a|D, to : smile|N, dis : 1, weight : 274.37365
from : beautiful|A, to : smile|N, dis : 0, weight : 240.716

Table 6: Example edges from sample phrase "with a beautiful smile"

2.3. Graph Based Contextual Similarity

Given a entry to normalize, next step is extracting normalization candidates for each OOV token with contextual similarity features. For each ill-formed OOV token in a given entry, we start with listing the related tokens in that entry. The list includes all the related words of an ill-formed OOV token and their distance to the OOV token. We will refer this list as neighbour list. In Table 7 you can find a sample neighbour list for the OOV token beautiful|A from the sample sentence in Table 4.

w|P, distance: 2
a|D, distance: 1
smile|V, distance: 1

Table 7: Example neighbour list for the beautiful|A

For each neighbour in the neighbour list we traverse the graph and find the edges from or to the neighbour. These edges (neighbour,cand) or

Candidate selection from graph

2.4. Lexical Similarity

dictionary from graph

slang dictionary lookup

double metaphone 1 edit distance 2

longest common sub-sequence ratio

2.5. Ranking

butun parametetreler ayn.Hassan'larn lambdas

[1] J. Si, A. Mukherjee, B. Liu, Q. Li, H. Li, X. Deng, Exploiting topic based twitter sentiment for stock prediction, in: ACL (2), The Association for Computer Linguistics, 2013, pp. 24–29.

[2] C. Danescu-Niculescu-Mizil, M. Sudhof, D. Jurafsky, J. Leskovec, C. Potts, A computational approach to politeness with application to social factors, in: [9], 2013, pp. 250–259.

[3] T. Sakaki, M. Okazaki, Y. Matsuo, Earthquake shakes twitter users: Real-time event detection by social sensors, in: Proceedings of the 19th International Conference on World Wide Web, WWW '10, ACM, New York, NY, USA, 2010, pp. 851–860. URL: `http://doi.acm.org/10.1145/1772690.1772777`. doi:`10.1145/1772690.1772777`.

[4] A. Ritter, C. Cherry, B. Dolan, Unsupervised modeling of twitter conversations, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 2010, pp. 172–180. URL: `\url{http://scholar.google.de/scholar.bib?q=info:5hTIjtmFJKAJ:scholar.google.com/&output=citation&hl=de&as_sdt=0&ct=citation&cd=67}`.

[5] B. Han, T. Baldwin, Lexical normalisation of short text messages: Makn sens a #twitter, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, HLT '11, Association for Computational Linguistics, Stroudsburg, PA, USA, 2011, pp. 368–378. URL: http://dl.acm.org/citation.cfm?id=2002472.2002520.

[6] H. Hassan, A. Menezes, Social text normalization using contextual graph random walks, in: [9], 2013, pp. 1577–1586.

[7] O. Owoputi, B. O'Connor, C. Dyer, K. Gimpel, N. Schneider, N. A. Smith, Improved part-of-speech tagging for online conversational text with word clusters, in: HLT-NAACL, The Association for Computational Linguistics, 2013, pp. 380–390.

[8] K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, N. A. Smith, Part-of-speech tagging for twitter: Annotation, features, and experiments, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2, HLT '11, Association for Computational Linguistics, Stroudsburg, PA, USA, 2011, pp. 42–47. URL: http://dl.acm.org/citation.cfm?id=2002736.2002747.

[9] conf/acl/2013-1, Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers, The Association for Computer Linguistics, 2013.