Whose line is it anyway?
Automatic Multilingual Emotion Annotation of Movie Dialogues

Wojciech Stokowiec
National Information Processing Institute
al. Niepodleglosci 188B
00-608 Warsaw
wstokowiec@opi.org.pl

Abstract

In this work we take advantage of both semantic networks provided by BabelNet [3] and distributed word representations trained by skip-gram model with negative sampling in an effort to obtain vector representations of 8 emotional states. Subsequently, we have employed abovemention emotion vectors to identify emotional scenes in several contemporary movies spanning different genres and languages (Polish and English).

1 Introduction

With the rise of popularity of movie streaming and the explosion of available content it is increasingly important to find the movies which fit the diverse tastes of the viewers. In this work we introduce an unsupervised, automatic emotion annotation algorithm, which could be used for video information retrieval and recommendation systems. It is straightforward, to further extend our approach for additional task such as movie clustering.

In our work we combine hypernym/hyponym relations and synonyms of BabelNet and distributed word representation, which allows us to quickly annotate movie dialouges by reducing network communication overhead. This approach is dictated by the fact that BabelNet API calls are relatively expensive and are prohibitive in production systems.

The source code for this demo is avialable on GitHub[1].

[1] https://github.com/1o0ko/emotion_vecotrs

2 Data Set

We based our analysis on OpenSubtitles2016[2] from the **OPUS**, a freely available, growing language resource of parallel corpora and related tools [4].

To obtain vector representation of words we trained skip-gram model [2] with negative sampling skip-gram [1] on english and polish Wikipedia.

3 Algorithm Description

In this section we will briefly describe proposed approach.

3.1 Emotion Vectors

In an effort to obtain emotion vectors we have used BabelNet to construct Sense Trees with synset as nodes. Each Sense Tree represents one of the eight basic emotional states ('*love*', '*happiness*', '*surprise*', '*emotionless*', '*sad*', '*disgust*', '*anger*' and '*fear*') as root and synsets of hyponyms as children. Depth of the Sense Tree has been decided to be equal to three. Subsequently, each node in the Tree has been assigned a vector representation of the synset's lemma. Finally, we have calculated weighted average of word vectors in the Sense Tree with weights proportional to the distance from the root. Detailed description of Sense Tree and Emotion Vectors construction can be find in the algorithm listings 1 and 2 respectively.

3.2 Sentence Vectors

For selected movies we have extracted both English and Polish subtitles and aligned them with the help of OpenSubtitles2016. Subsequently, for each line of dialogue we removed stop words and constructed sentence vectors by averaging word

[2] a collection of translated movie subtitles from http://www.opensubtitles.org/

Function `AddChildren` (*root, level*)
  input: A root *Sense* object *root*,
    A level a of recursion *level*
  if *level* < *maximum level* **then**
    $hs \leftarrow$ `GetHyponyms` (*root*) **for** $h \leftarrow\in hs$ **do**
      if *h.weight* > *threshold* **then**
        root.addChild(h)
      end
    end
    for $ch \in root.children$ **do**
      `AddChildren` (*child, level* + *1*)
    end
  end

Algorithm 1: Sense Tree Construction

vectors of the remaining words. Finally, we assigned the closest emotion vector to the sentence vector using cosine distance as a measure.

Function `GeVector` (*tree, level*)
  input : A *Sense Tree* object *tree*
  output: Vector representaion of a tree
  $v \leftarrow$ `WordToVec` (*tree.lemma*)
  $w \leftarrow 1$ if tree is a leaf, the inverse of the number of children otherwise.
  if *tree has children* **then**
    for $ch \in root.children$ **do**
      $v \leftarrow v + w*$ `GeVector` (*ch, level* + *1*)
    end
  end
  return $v/level$

Algorithm 2: Emotion Vector Construction

4 Results

Since subtitles with labeled emotions were not available, we decided to test our approach by plotting distribution of emotions for given periods of time for both languages and comparing them visually.

Plots were constructed in a following way: for a given movie, we grouped all dialogues in 10 sentences chunks and calculated distribution of emotion states for a given group. As a result, for each ANALYSED movie, we obtained 8 dimensional time series. Results are shown on figures 1, 2 and 3. Courious reader can comparision of emotion for Cassablanca and Pulp Fiction in table 1

Table 1: Compatision of emotions

	Casablanca		**Pulp Fiction**	
	EN	**PL**	**EN**	**PL**
love	851	116	1210	112
sad	42	113	33	78
emotionless	133	694	121	722
disgust	33	302	17	407
anger	56	94	61	58
surprise	320	94	157	234
fear	241	128	168	123
happiness	68	87	50	83
sum	**1744**	**1744**	**1817**	**1817**

5 Conclusions and Further Work

Although, for some sentences algorithm is able to accurately match the appropriate emotion class, it seems that in general the results are not stable across languages. For each English movie we have evaluated our algorithm on, the most common emotion was *love*. In Polish language however the it was 'emotionless' that dominated the movie.

It would be interesting take the advantage of hyponym/hypernym relations for measuring the distance between a specific emotional word and the emotional concept with which it is associated (if it is).

References

[1] Tomas Mikolov et al. "Distributed Representations of Words and Phrases and their Compositionality". In: *Advances in Neural Information Processing Systems 26*. 2013, pp. 3111–3119.

[2] Tomas Mikolov et al. "Efficient Estimation of Word Representations in Vector Space". In: *CoRR* abs/1301.3781 (2013).

[3] Roberto Navigli and Simone Paolo Ponzetto. "BabelNet: The Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network". In: *Artificial Intelligence* 193 (2012), pp. 217–250.

[4] Jog Tiedemann. "Parallel Data, Tools and Interfaces in OPUS". In: *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. May 2012.

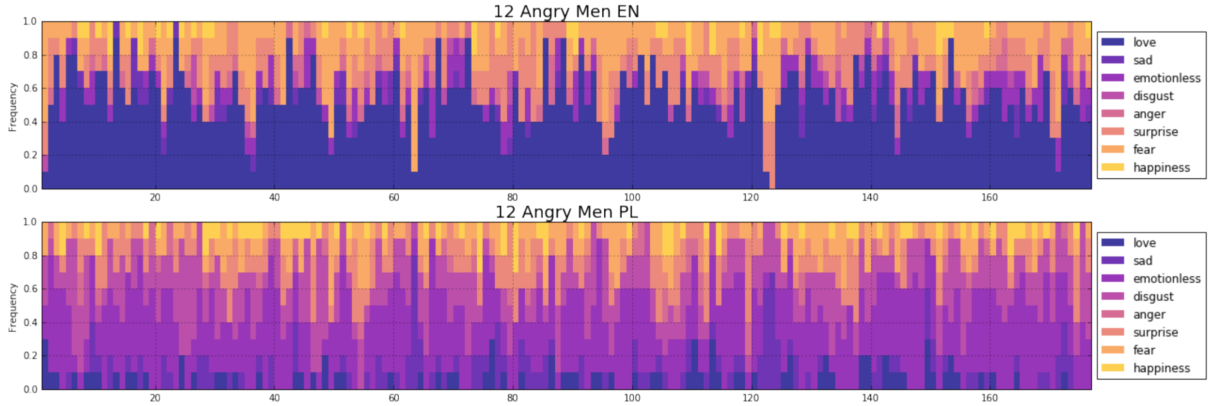

Figure 1: Distribution of emotions over time in '12 Angry Men'

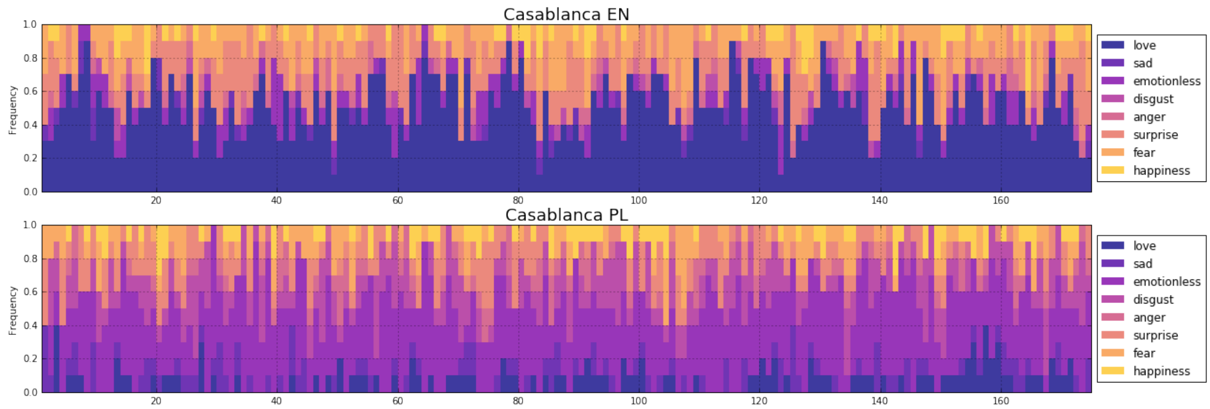

Figure 2: Distribution of emotions over time in 'Cassablanca'

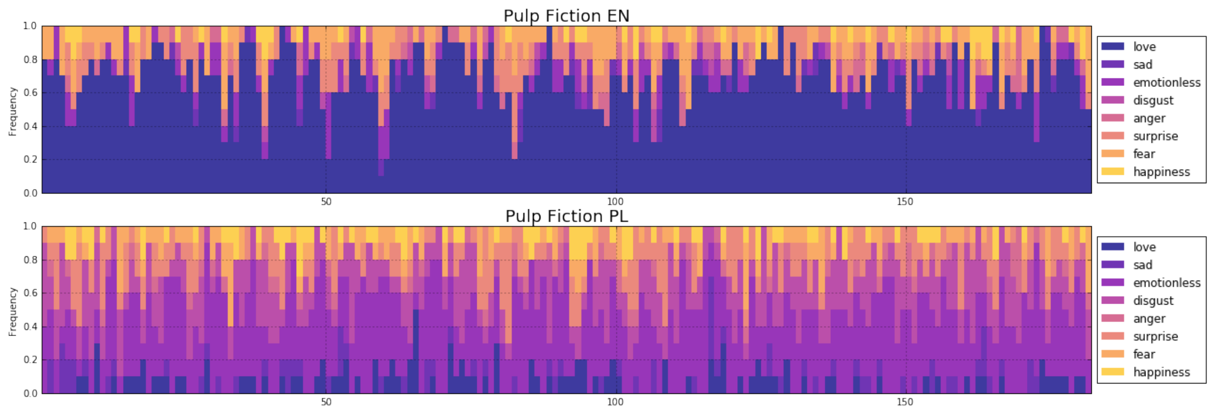

Figure 3: Distribution of emotions over time in 'Pulp Fiction'