# Detecting New Word Meanings: A Comparison of Word Embedding Models in Spanish

**Andrés Torres Rivera** [1,2], **Juan-Manuel Torres-Moreno** [2,3]

1. *Univrersitat Pompeu Fabra*
   *Roc Boronat 138*
   *Barcelona, Spain*
   *andres.torres@upf.edu*
2. *Université d'Avignon*
   *74 Rue Louis Pasteur, 84029*
   *Avignon, France*
   *juan-manuel.torres@univ-avignon.fr*
3. *Polytechnique Montréal*
   *C.P. 6079, succ. Centre-ville*
   *Montréal (Québec) Canada H3C 3A7*

ABSTRACT. *Semantic neologisms (SN) are defined as words that acquire a new word meaning while maintaining their form. Given the nature of this kind of neologisms, the task of identifying these new word meanings is currently performed manually by specialists at observatories of neology. To detect SN in a semi-automatic way, we developed a system that implements a combination of the following strategies: topic modeling, keyword extraction, and word sense disambiguation.*
*The role of topic modeling is to detect the themes that are treated in the input text. Themes within a text give clues about the particular meaning of the words that are used, for example: viral has one meaning in the context of computer science (CS) and another when talking about health. To extract keywords, we used TextRank with POS tag filtering. With this method, we can obtain relevant words that are already part of the Spanish lexicon. We use a deep learning model to determine if a given keyword could have a new meaning. Embeddings that are different from all the known meanings (or topics) indicate that a word might be a valid SN candidate. In this study, we examine the following word embedding models: Word2Vec, Sense2Vec, and FastText. The models were trained with equivalent parameters using Wikipedia in Spanish as corpora. Then we used a list of words and their concordances (obtained from our database of neologisms) to show the different embeddings that each model yields. Finally, we present a comparison of these outcomes with the concordances of each word to show how we can determine if a word could be a valid candidate for SN.*

*RÉSUMÉ. Les néologismes sémantiques (NS) sont définis comme des mots qui acquièrent une nouvelle signification tout en maintenant leur forme. Compte tenu de la nature de ce type de néologisme, la tâche d'identifier ces nouveaux sens des mots est actuellement effectuée manuellement par des spécialistes des observatoires de néologie. Pour détecter les NS de manière semi-automatique, nous avons développé un système mettant en uvre une combinaison des stratégies suivantes: modélisation de sujets, extraction de mots-clés et désambiguïsation du sens des mots. Le rôle de la modélisation des sujets est de détecter les thèmes traités dans le texte dentrée. Les thèmes d'un texte donnent des indications sur le sens particulier des mots utilisés. Par exemple, viral a un sens dans un contexte informatique (CS) et un autre lorsqu'il est question de santé. Pour extraire des mots-clés, nous avons utilisé TextRank avec filtrage des balises POS. Avec cette méthode, nous pouvons obtenir des mots pertinents qui font déjà partie du lexique espagnol. Nous utilisons un modèle d'apprentissage profonde pour déterminer si un mot-clé donné peut avoir une nouvelle signification. Des word embbeding différentes de toutes les significations connues (ou sujets) indiquent qu'un mot peut être un candidat NS valide.*

*Dans cette étude nous examinons les modèles word embbeding suivants: Word2Vec, Sense2Vec et FastText. Les modèles ont été formés avec des paramètres équivalents en utilisant Wikipédia en espagnol en tant que corpus. Nous avons ensuite utilisé une liste de mots et leurs concordances (obtenus à partir de notre base de données de néologismes) pour montrer les différentes imbrications générées par chaque modèle. Enfin, nous présentons une comparaison de ces résultats avec les concordances de chaque mot pour montrer comment nous pouvons déterminer si un mot peut être un candidat valide pour NS.*

*KEYWORDS: Word Embedding, Deep Learning, Neology, Natural Language Processing, Machine Learning, Terminology*

*MOTS-CLÉS : Word Embedding, Apprentissage Profonde, Néologie, Traitement Automatique des Langues, Apprentissage automatique, Terminologie*

DOI:10.3166/RIA.XX.1-17 

## 1. Introduction

The detection of semantic neologisms (SN) is a task that is currently performed by specialists at observatories of neology such as OBNEO[1], NEOPORTERM[2] and OBNEQ[3]. This task usually consists in reading and analyzing newspapers to highlight ambiguous words that might be valid candidates for SN. A valid candidate is defined as a known word that, in a certain context, acquires a new meaning, usually a meaning related to the subject that is being treated. For further delimitation of this phenomenon we refer to the classification of neologisms proposed by Cabré (2009): a semantic neologisms is a type of neologism that goes trough a semantic process which could reduce, expand, or change its meaning.

Table 1: Excerpt of the multivariable table (Cabré, 2009, p. 35)

| Process | Formation | Grammatical change | Category (FCONV) | |
|---|---|---|---|---|
| | | | **Subcategory (SINT)** | |
| | | | **Semantic (re-semantization)** | Meaning reduction<br>Meaning expansion<br>Meaning change |

The scope of this study is limited to SN that have their origin as specialized knowledge units (terms) as defined by Cabré and Estopà 2005:

> [...]a lexical unit, whose structure is related to an origin lexical unit or product of the lexicalization of a syntagm, that has a specific meaning in the ambit to which it is related and it is necessary in the conceptual structure of the domain in which it takes part. (Cabré, Estopà, 2005, p.77)

These terminological units are part of a broader general concept defined as specialized significance units (SSU[4]). There are different types of SSU, such as lexical, verbal, nominal, adjectival or adverbial (Estopà, 2013). In this study we will explore verbal, nominal and adjectival SSU obtained from our SN database. To carry out the detection of SN in a semiautomatic way, we designed a system that uses topic detection, keyword extraction and word embeddings to detect new word meanings. We named this system DENISE[5], a multilingual tool that mimics the work that specialists perform at OBNEO.

1. Observatori de Neologia (OBNEO) https://www.upf.edu/web/obneo/
2. Observatório de Neologia e de Terminologia em Língua Portuguesa (NEOPORTERM) http://clunl.fcsh.unl.pt/investigacao/projetos-concluidos/neoporterm-observatorio-de-neologia-e-de-terminologia-em-lingua-portuguesa/
3. Observatoire de Néologie du Québec (OBNEQ) http://www.lli.ulaval.ca/recherche/groupes-et-laboratoires
4. In Catalan: *Unitats de Significació Especializada.*
5. A play on words that fits its three working languages: *detector de neolgoismos semánticos*, in Spanish; *detector de neologismes semàntics*, in Catalan and *détecteur de néologismes sémantiques* in French.

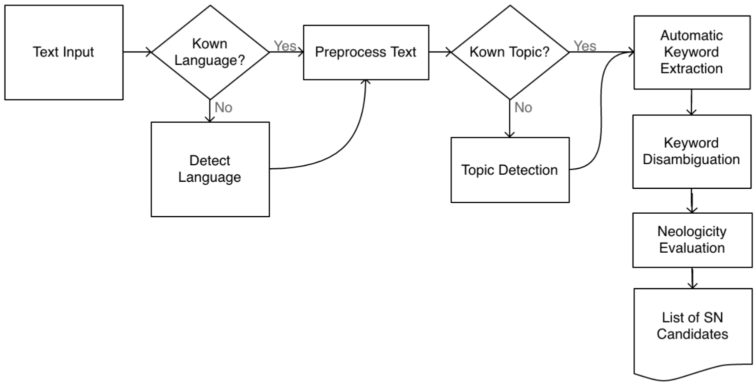


Figure 1: Work-flow Diagram of the DENISE System

DENISE takes simple text as input, then the user can set the working language or let the system detect it automatically, currently the supported languages are: Catalan, French and Spanish. The next step is text preprocessing: normalization and tokenization as well as elimination of stopwords, non latin characters and symbols. Once the text is preprocessed the user has the option to introduce the theme of the text or let the system detect the topics within the text using a TF-IDF and logistic regression based model. Once the text has a topic assigned, the system will automatically extract keywords using TextRank with POS filtering.

| Àmbit geogràfic | D. Publicaci | Entrada | Categoria gramatical | Context |
|---|---|---|---|---|
| COLOMBIA | 6/7/14 | agache | m (Nom masculí) | Nunca olvidaré como periodista el dolor de saber que la Selección Colombia había clasificado al mur |
| CHILE | 26/2/08 | apalancado -da | adj (Adjectiu) | Una de las mayores preocupaciones es la amplia exposición de Goldman a los préstamos *apalancado |
| PERÚ | 9/3/08 | fuera de juego | loc (Locució) | El número uno de las FARC es un hombre que está enfermo y debido a su situación física prácticamen |
| CHILE | 20/2/08 | pingüino -na | adj (Adjectiu) | Respecto a Traverso, afirmó haber tenido una buena relación con él, marcada por las complejidades |
| CUBA | 4/3/04 | a bordo | loc (Locució) | Rey Isaac, con uno *a bordo*, y Rolando Meriño, con otros dos en los senderos, así como el granmen |
| URUGUAY | 3/1/04 | a costillas | loc (Locució) | Además de playa y sol ¿qué más necesita Uruguay para ser un país de turismo? Que no usemos la vive |
| PERÚ | 1/3/04 | a diestra y siniestra | loc (Locució) | Los partidarios el presidente saliente asaltaron bancos, tiendas y gasolineras, mientras disparaban *a |
| ESPAÑA (Barcelona) | 17/5/97 | a escala | loc (Locució) | Aunque ambos líderes mantendrían la negociación *a escala parlamentaria, antes de la votación de e |
| PERÚ | 13/8/08 | a la champa | loc (Locució) | Los fujimoristas plantean que la Comisión de Fiscalización cite al ministro de Vivienda y Construcció |
| PERÚ | 19/7/12 | a las patadas | loc (Locució) | Consultada sobre las críticas al gobierno por el manejo de los conflictos sociales, subrayó que se trab |
| CATALUNYA (Barcelona) | 1/1/05 | cara | f (Nom femení) | Todas las *caras* de Jim Carrey. |
| CATALUNYA (Barcelona) | 1/1/05 | girar | v intr (Verb intransitiu) | Hicimos con él un programa de TV en Inglaterra y funcionó tan bien que estamos muy emocionados |
| MÈXIC | 21/2/09 | a punta de | loc (Locució) | Como saga del colectivo tijuanense que ha brillado al combinar beats digitales con teclados análogo |
| MÈXIC | 27/4/11 | abanicar | v intr (Verb intransitiu) | Incluso, reprochó que mientras el PRI impulsa la reforma, yo veo al PAN muy calladito, ni se despein |
| MÈXIC | 7/6/14 | abatir | v tr (Verb transitiu) | Elementos de las fuerzas federales *abatieron* a cinco presuntos delincuentes en la carretera Tampic |
| ESPAÑA (Barcelona) | 14/12/97 | abducción | f (Nom femení) | Por supuesto, también habrá referencias a la *abducción extraterrestre de la hermana de Mulder, pa |
| ESPAÑA (Barcelona) | 1/1/05 | modigliani | m (Nom masculí) | Los picassos y los *modiglianis* originales estarán enseguida por los suelos porque todo el mundo qu |
| CATALUNYA (Barcelona) | 1/1/05 | secuela | f (Nom femení) | La comedia, *secuela* de Los padres de ella, está protagonizada nuevamente por Robert De Niro y Be |
| ESPAÑA (Barcelona) | 14/6/91 | abierto | m (Nom masculí) | Al menos una persona resultó muerta y otras cinco heridas a causa de un rayo caído sobre los especta |

Figure 2: Excerpt from the OBNEO Database

Each of these keywords serve as query items for our word embeddings models; with these queries we can obtain the 140 most similar terms[6] and then create a "semantic field" (SF) for each keyword. This SF serves as a representation of the most common meaning of each queried word in the general language. Then, the system

6. We set topn=140 because the input texts are short concordances, varying from 130 to 150 characters each after filtering concordances with non-informative contexts.

evaluates these SF to detect their theme and proceeds to evaluate if concordance exists between the topic of the input text and the topic of the SF. To obtain valid candidates for SN, DENISE filters the keywords keeping only keywords that meet the following criteria: a candidate for SN is a keyword whose detected topic is different from the topic of the embeddings, since this would indicate that a known word is being used is a context that is different from its most common SF.

## 2. Related Work

The automatic detection of SN, because of its nature, has been a more complicated task compared with other kinds of neologisms such as transferred words or derived words (Tebé, 2002; Janssen, 2009; Renouf, 2010; Sablayrolles, 2012; Reutenauer *et al.*, 2011). Therefore, presently there is a necessity that is being covered partially. One of the first specialized systems to detect SN is April (Renouf, 1998, 2010, 2012); this approach uses statistical and linguistic rules, collocation patterns and heuristic rules to track semantic change over time. These methods include boot-strapping, a chronologically divided corpus used as a control database and a reference dictionary. April analyzes common collocations to track SN, which means that a word that a appears in a context different from its most usual context might be a valid candidate for SN. However, no evaluation is provided and the authors mention two specific problems: the superficial definition of novelty does not distinguish between a new sense or a new reference, and April can not identify collocations that have at least four concordances.

A second system that could be used to detect SN is Logoscope[7] (Falk, Bernhard, Gérard, 2014b,a; Falk, Bernhard, Gérard, Potier-Ferry, 2014; Gérard *et al.*, 2014; Bernhard *et al.*, 2015a,b), which uses a combination of topic modeling using Latent Dirichlet Allocation (LDA) and a linear support vector machine classifier that could be used to identify possible new word meanings. In order to detect SN, Logoscope analyzes theme concordance between the collocation of a keyword and its definition in the dictionary: when the collocation and the definition do not share the same topic this could indicate that a given keyword might be candidate for SN. The authors were able to detect a new sense for the word *quenelle* using this methodology, but they affirm that relying on dictionary definitions complicates this task given the nature of SN. While the authors provide all the formulas that were used, there is no formal evaluation for this particular task.

Finally, there are other methodological approaches such as those proposed by Janssen (2005, 2009, 2012), who developed a POS tagger that uses statistical rules and could obtain SN as a byproduct from this process: ambiguous words are assigned a special tag that, when inspected, can indicate a semantic change in course. Finally, Nazar (2011, 2013, 2014) proposes a methodology that involves the combination of word sense induction (WSI) and clustering to group word senses. The author states that this method could be implemented to develop a tool that detects SN automatically.

7. http://logoscope.unistra.fr

While both methodologies explain how word meanings can be grouped and classified, neither provide implementations for the detection of SN.

## 3. Theme Detection

As part of our methodology, we expect to classify the different themes or topics that are being treated in the input text, because we assume that new word meaning depends on the topic in which we find this word. For example, if a word has one known meaning related to economy and we find this same word in a CS text, this word might have a change of meaning related to this new theme. We treated this step as a classification problem, which means that each text that is entered to DENISE is evaluated using a logistic regression to predict its main theme or topic.

Our corpus was compiled using articles from specialized publications in Spanish: PC World (CS) with a total of 308,930 words; Marca (sports), 275,872 words; and El Financiero (economy), 280,404 words. This corpus was used to generate a TF-IDF model and then train the logistic regression model using the following parameters: L2 penalty, max intercept scaling of 1, max tolerance of 1e-4 and 1.0 for inverse of regularization strength, the resulting confusion matrix can be seen in Figure 3.

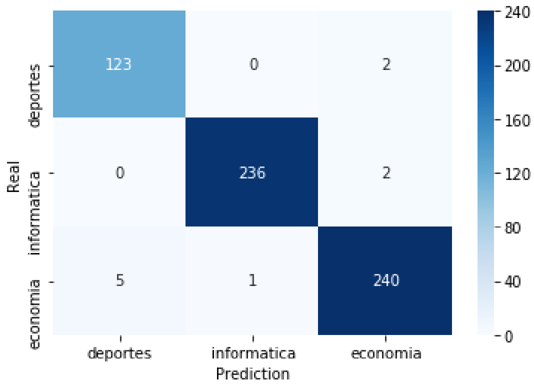


Figure 3: Confusion Matrix of the Logistic Regression Model

The model is capable of detecting three topics: sports (*deportes*), economy (*economía*), and computer science (*informática*). As we mentioned before, our goal is to detect new word meanings related to computer science, therefore we expect that, for every text we evaluate, DENISE will return “informática” (labeled as 1) as the main topic, and if it does not detect CS as the main topic it will return “not informática” (labeled as 0). We compared the precision of three classifiers using a cross validation score: logistic regression obtained 0.982; multinomial naïve Bayes, 0.898; and random forest classifier, 0.790. After obtaining these results, we selected the logistic regression model and proceeded to evaluate its mean accuracy on a train-test split: it obtained 0.913 on the train set and 0.889 on the test set.

## 4. Keyword Extraction

After detecting the theme of the input text, DENISE extracts the keywords that will be evaluated as candidates. One of the particularities of SN is that they are known words and, therefore, we can not use a set of lexicographical rules or dictionaries because we are not looking for new words at a formal (structural) level, but for new meaning of known words. For this reason we decided to use the TextRank (Mihalcea, Tarau, 2004) algorithm (as defined by Equation 1) with POS tag filtering. This graph-based algorithm was inspired by the PageRank (Page *et al*., 1999) algorithm originally used by the Google search engine. TextRank is a widely used in ranking and recommendation systems, keyword extraction and automatic summarization systems (Li, Wang, 2014; Barrios *et al*., 2016; Pay *et al*., 2018).

$$WS(V_i) = (1-d) + d \times \sum_{V_j \in In(V_i)} \frac{w_{ij}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j) \tag{1}$$

DENISE's implementation of TextRank uses the original graph evaluation and incorporates POS tag filtering to prioritize the extraction of verbs, nouns and adjectives. These type of units are of interest because in our database of neologisms we found that most SN, as shown in Table 2, fall into these POS categories. With the implementation of a POS filter we obtained 14% more accuracy in comparison with the regular TextRank implementation, this might indicate that the algorithm is correctly extracting possible candidates.

Table 2: NS Distribution by Part of Speech

| POS | Total |
|---|---|
| adj (Adjective) | 750 |
| adv (Adverb) | 3 |
| conj (Conjunction) | 2 |
| f (Feminine Noun) | 1280 |
| f pl (Feminine Plural Noun) | 20 |
| interj (Interjection) | 3 |
| loc (Locution) | 85 |
| m (Masculine Noun) | 2425 |
| m i f (F. and M. Noun) | 61 |
| m pl (Plural Masculine Noun) | 62 |
| n (Neutral Noun) | 3 |
| pron (Pronoun) | 2 |
| v intr (Intransitive Verb) | 146 |
| v pron (Pronominal Verb) | 129 |
| v tr (Transitive Verb) | 591 |

## 5. Word Embedding Models

The last step in DENISE's analysis process is sense disambiguation using word embeddings. With the resulting keyword from the previous step, our work hypothesis is the following: a new word sense might be found when a known word is used in a text about a topic that is different from the topics where this word is usually collocated. Therefore we assume that the most common word representation of a given keyword is closely related to its main (or most common) meaning, and this meaning might is also related to a certain topic.

We carried out an analysis using three different neural network based models: Word2Vec (Mikolov, Chen *et al*., 2013; Mikolov, Sutskever *et al*., 2013), FastText (Bojanowski *et al*., 2016; Joulin *et al*., 2016) and Sense2Vec (Trask *et al*., 2015). To train the models we used Wikipedia in Spanish as corpus and the training was performed using the same training values described in the bibliography. In the following subsections we describe some of the particularities of each model and the training parameters that were used.

**Word2Vec**. Word2Vec is a model that uses neural networks to produce dense vector representations of words. Its two main architectures are skip-gram and CBOW, the first one being slower but better for projecting uncommon words. In order to perform the disambiguation process, we require embeddings that represent the most common meaning of the input keywords, therefore we trained our skip-gram based model with a dimension size of 300, a window of 5 and a min count of 20 elements.

**Sense2Vec**. To obtain a Sense2Vec model we tagged the same Wikipedia corpus that was used to train the Word2Vec model, using the Universal Dependencies[8] tagset. This approach allows for the generation of a model that has similar characteristics to a Word2Vec model with the added advantage of POS tag disambiguation. This means that a word that is being used as a noun and a verb has two different representations in the model, one for each case. We followed the training parameters proposed by Trask *et al*. (2015): a dimension size of 500, a window of 5 and a min count of 10 elements, again using continuous skip-grams.

**FastText**. Altough the FastText model also uses neural networks to generate word representations, this model uses subword data as the minimum units to train these representations; meanwhile, the minimum unit to trained by Word2Vec and Sense2Vec are words. For example, in a FastText model the vector for the word "viral" would be composed by the ngrams within "viral" in the following way: "<vi", "vir", "vira", "viral", "viral>", "ira", "iral>", "iral>", "ral", "ral>", "al>". We used the pretrained vectors available at the FastText website[9], which

8. http://universaldependencies.org/u/pos/
9. https://fasttext.cc/docs/en/crawl-vectors.html

were trained using CBOW with position-weights, in dimension 300, with character n-grams of length 5, a window of size 5 and 10 negatives.

## 6. Pipeline Description

To extract SN from a text first our algorithm (3) detects the working language, if the language is not supported it continues with a new text. If the language is supported then DENISE proceeds to assign a theme to the input text using the logistic regression model. If the user knows the language and the topic it can be set manually, the number of keywords to extract and the number of words for the SF can also be set manually.

The next step is to extract keywords using TextRank, the implementation returns a list with the $n$ number of KW that were selected by the user. For each of the items from the TextRAnk list, the algorithm returns a SF composed by the TOPN most similar words for each item of the list using the word embedding model, which by default is set to Sense2Vec.

Finally, the systems evaluates if there is theme concordance between the semantic field and the input text. When concordance exists it indicates that the most common meaning from the embedding model is being use, which means that a given word is not a candidate for SN. Meanwhile, if concordance does not exist between both topic, this might indicate that a given keyword is a possible candidate for SN, when this is the case, the algorithm returns a list with the possible candidates and their detect themes so the user can determine if there are true SN candidates.

Table 3: Pseudocode Description

```
def sn_classification(text, lang='auto', topic='auto', KW=10, TOPN=140):
    if lang == 'auto':
        lang_flag = lang_detection(text)
    else:
        lang_flag = True, lang

    if lang_flag[0] == True:
        sn_list = []
        if topic == 'auto':
            text_topic = analyze_topic([text], lang_flag[1])
        else:
            text_topic = topic
        text_rank = extract_keywords(text, lang_flag[1], KW)
        for i in text_rank:
            sf = model.most_similar(i, TOPN)
            sf_topic = analyze_topic([sf], lang_flag[1])
            if text_topic != sf_topic:
                sn_list.append([i, text_topic, sf_topic])
            else:
                continue
        return sn_list
    else:
        print("Language not supported")
```

## 7. Model Comparison

To determine the performance of DENISE with each model, we used a list of keywords and their concordances (all pertaining to the CS field) from the SN database as input text. Our database has a total of 5562 SN registered and manually validated; from this total we selected SN that correspond to the CS field and discarded concordances shorter than 130 words to have enough context words. These are SN that have already been manually analyzed and all of them belong to the field of the CS.

With these keywords and concordances we generated a CSV table containing the 125 term-concordance items, then we proceeded to query each item of the list on each model to obtain the 140 most similar terms for each item. As shown in Table 4, FastText was the model that yielded the best results, since with this model we obtained 125 SF for each of the 125 terms that were queried, followed by the Word2Vec model with 100 from 125 terms, and last the Sense2Vec model with 97 of 125.

Table 4: Items obtained for each model

| Model | Expected | Recovered | Percentage |
|---|---|---|---|
| FastText | 125 | 125 | 100% |
| Word2Vec | 125 | 100 | 80% |
| Sense2Vec | 125 | 97 | 77.6% |

After obtaining the most similar terms for each item, we performed the topic detection process on these SF. This process is performed in order to assess if the generated SF have the same topic than the concordances. If the set of embeddings and the concordance share the same topic, it might indicate that the keyword in question might no be candidate for SN, whereas a set of dissimilar embeddings and concordance might indicate that the keyword is a candidate for SN. The results of this process are shown in Tables 5 and 6.

Table 5: Results of FastText and Word2Vec Models

| Model | SF | CS Label | Man. Eval. | Total Correct |
|---|---|---|---|---|
| FastText | 125 | 38 | 78 | 77 |
| Word2Vec | 100 | 34 | 57 | 63 |

While the Sense2Vec model was the model that retrieved the lowest number of unique terms, it provided the more representations in total: 172 when combining all three categories. This gives us more detailed information regarding use: a word might be used mainly as a verb but it could also be used a noun, and one of these uses could be the neological meaning.

We evaluated each of the SF manually to ensure that the automatic topic detection process was accurate, and to ensure that the model classifies the terms correctly: whether they belong to CS or not. After manually evaluating all the embeddings, we proceeded to evaluate the number of correct cases, that is, if the predicted topic is

Table 6: Results of the Sense2Vec Model

| Tag | SF | CS Label | Man. Eval. | Total Correct |
|---|---|---|---|---|
| VERB | 35 | 7 | 15 | 25 |
| NOUN | 87 | 32 | 44 | 73 |
| ADJ | 50 | 17 | 26 | 41 |

the same as the manually observed topic and the percentage of agreement between the automatically and the manually labeled SF. We expected that the classifier could determine if a context and a SF are, in fact, related to the CS field or not. We used f1-Score, precision, recall and support for each model; in the case of the Sense2Vec model we calculated these metric for each POS category independently.

The results for the FastText model can be seen in Table 7 and the results for Word2Vec on Table 8. Both models obtained similar f1-scores, and both seem to have high precision and low recall when classifying SF that do not belong to CS, but also present low precision and high recall when classifying SF that belong to CS.

Table 7: FastText classification results

| | f1-score | precision | recall | support |
|---|---|---|---|---|
| 0 | 0.641791 | 0.914894 | 0.494253 | 87.0 |
| 1 | 0.586207 | 0.435897 | 0.894737 | 38.0 |
| micro avg | 0.616000 | 0.616000 | 0.616000 | 125.0 |
| macro avg | 0.613999 | 0.675396 | 0.694495 | 125.0 |
| weighted avg | 0.624893 | 0.769279 | 0.616000 | 125.0 |

Table 8: Word2Vec classification results

| | f1-score | precision | recall | support |
|---|---|---|---|---|
| 0 | 0.660550 | 0.837209 | 0.545455 | 66.0 |
| 1 | 0.593407 | 0.473684 | 0.794118 | 34.0 |
| micro avg | 0.630000 | 0.630000 | 0.630000 | 100.0 |
| macro avg | 0.626979 | 0.655447 | 0.669786 | 100.0 |
| weighted avg | 0.637722 | 0.713611 | 0.630000 | 100.0 |

On the other hand, the Sense2Vec model obtained better f1-scores than the Word2Vec and FastText model, with being NOUN the most productive –and balanced– category with a weighted average f1-score of 0.84. This value is also greater than both the f1-score of the Word2Vec model and the FastText model. Overall, while the Sense2Vec model retrieved the least amount of SF, the resulting embeddings were better classified.

Finally, following the condition of disagreement between the topic of the embeddings and the topic of the input text, each model generated a list o candidates for SN. The Sense2Vec model generated a list of 55 candidates from the original 125 SN

Table 9: Sense2Vec classification results

| | f1-score | precision | recall | support |
|---|---|---|---|---|
| **Verbs** | | | | |
| 0 | 0.791667 | 0.950000 | 0.678571 | 28.0 |
| 1 | 0.545455 | 0.400000 | 0.857143 | 7.0 |
| micro avg | 0.714286 | 0.714286 | 0.714286 | 35.0 |
| macro avg | 0.668561 | 0.675000 | 0.767857 | 35.0 |
| weighted avg | 0.742424 | 0.840000 | 0.714286 | 35.0 |
| **Nouns** | | | | |
| | f1-score | precision | recall | support |
| 0 | 0.857143 | 0.976744 | 0.763636 | 55.0 |
| 1 | 0.815789 | 0.704545 | 0.968750 | 32.0 |
| micro avg | 0.839080 | 0.839080 | 0.839080 | 87.0 |
| macro avg | 0.836466 | 0.840645 | 0.866193 | 87.0 |
| weighted avg | 0.841932 | 0.876625 | 0.839080 | 87.0 |
| **Adjectives** | | | | |
| | f1-score | precision | recall | support |
| 0 | 0.842105 | 1.000000 | 0.727273 | 33.0 |
| 1 | 0.790698 | 0.653846 | 1.000000 | 17.0 |
| micro avg | 0.820000 | 0.820000 | 0.820000 | 50.0 |
| macro avg | 0.816401 | 0.826923 | 0.863636 | 50.0 |
| weighted avg | 0.824627 | 0.882308 | 0.820000 | 50.0 |

list, FastText 42 candidates and Word2Vec, 35 candidates. The lists that each model generated are shown below:

**Sense2Vec Candidates:** 'almacenado', 'navegabilidad', 'palm', 'mini', 'cablear', 'controladora', 'terminal', 'viral', 'descarga', 'navegación', 'cargarse', 'objeto', 'cuenta', 'perfil', 'visual', 'directorio', 'asistente', 'bitácora', 'acelerar', 'chip', 'caída', 'caerse', 'conversión', 'muro', 'word', 'cortafuego', 'vacuna', 'nube', 'infectar', 'celular', 'gusano', 'troyano', 'dominio', 'navegar', 'alojamiento', 'electrónico', 'portal', 'migración', 'aplicación', 'ipod', 'motor', 'procesador', 'agujero', 'avatar', 'androide', 'piratería', 'virus', 'enlace', 'apuntador', 'subir', 'clonación', 'vínculo', 'api', 'herramienta', 'guru'.

**FastText Candidates:** 'almacenado', 'navegabilidad', 'palm', 'jaquear', 'controladora', 'game boy', 'navegación', 'cargarse', 'iserie', 'cuenta', 'clonar', 'visual', 'menú', 'asistente', 'acelerar', 'caída', 'caerse', 'conversión', 'mapeo', 'muro', 'vacuna', 'nube', 'infectar', 'gusano', 'dominio', 'navegar', 'alojamiento', 'correo_electrónico', 'electrónico', 'migración', 'clic', 'motor', 'agujero', 'avatar', 'virus', 'apuntador', 'subir', 'vínculo', 'disco duro', 'descargar', 'guru', 'descargarse'.

**Word2Vec Candidates:** 'almacenado', 'navegabilidad', 'palm', 'jaquear', 'viral', 'parche', 'navegación', 'cargarse', 'cuenta', 'clonar', 'visual', 'asistente', 'acelerar', 'caída', 'caerse', 'conversión', 'muro', 'word', 'vacuna', 'nube', 'infectar', 'troyano', 'gusano', 'dominio', 'navegar', 'alojamiento', 'migración', 'motor', 'agujero', 'avatar', 'piratería', 'virus', 'subir', 'vínculo', 'guru'.

## 8. Discussion

It is a common practice to assume that word representations created using the methods mentioned above can give useful information to create NLP applications. Nevertheless, upon manually analyzing all the resulting SF, we observed that some representations are ideal, ambiguous, represented in an foreign language (L2) used inside the working language (L1) and non-informative. Some examples of the last three groups include *nube* (*cloud*) and *dominio* (*domain*) from FastText; and palm from Word2Vec.

Table 10: Different Types of Word Representations

| Non Informative | Ambiguous | L2 in L1 |
|---|---|---|
| Dominio | Nube | Palm |
| "Dominio" | "cloudcomputing" | "plum" |
| "dominios" | "OwnCloud" | "frog" |
| "dominio.El" | "SoftLayer" | "wood" |
| "dominio." | "ownCloud" | "leaved" |
| "eldominio" | "IaaS" | "lily" |
| "sub-dominio" | "nubecitas" | "ferns" |
| "dominio.En" | "neblina" | "oak" |
| "dominio.-" | "virtualizada" | "apple" |
| "dedominio" | "NUBE" | "maple" |
| "domino" | "SaaS" | "gum" |
| "subdominio" | "hiperconvergencia" | "found_in" |
| "dominiode" | "niebla" | "native" |
| "dominio.La" | "clouds" | "leaf" |
| "dominio.com" | "Wordle" | "fruit" |
| "dominio-" | "vaporosa" | "jelly" |

In the case of DENISE, the use of a different method of classification ensures that the data goes trough a double-check step that turns in candidates that otherwise would be discarded. As a general recommendation, the linguistic content should be taken into account when implementing neural word embeddings. Regarding the particularities of each model, one key disadvantage of Word2Vec (specially for this task) is that it only yields representations of one meaning of the words that conform the vocabulary, and, as a consequence, there are other known meanings that are not being represented in this model. This kind of modeling could create ambiguous embeddings.

The overall performance of FastText was adequate. Even thought it might not be useful for this particular task, this kind of model might be better suited for detecting new words on a formal level, since it creates words representations for words that are not included in the vocabulary. This model could also be useful for analyzing composition and derivation processes on a lexical and morphological level.

The Sense2Vec model gave the best results for this particular task, in great part due to implementation of POS tags. These tags add information that can be used to disambiguate meaning of new words or polysemic words. However, from the 125 keywords it only had representations for 97. This might be due to the training parameters suggested by the authors or that, in comparison with FastText, we require more training data. Wikipedia is commonly used as corpus, but for a system that requires a general and broad representation of a language, more diverse data is required.

## 9. Conclusions and Future Work

In this study we have shown the application of word embeddings for the detection of semantic neologisms. For this particular task, the Sense2Vec model gave the best performance. We explored some of the advantages that FastText models have over Word2Vec; for instance, representations of uncommon words.

After further manual analysis of the most similar terms that each models generated, we observed three types of representations that are not useful for the development of the DENISE system: ambiguous embeddings, L2 in L1 embeddings and non informative embeddings. These kind of embeddings should be taken into account when designing an NLP application since the final goal is to implement rich linguistic knowledge. In the case of DENISE, we use TF-IDF and logistic regression for theme classification so the system does not rely on one single method to analyze semantic change.

While analyzing the characteristics of the generated embeddings we could observe that ambiguous representations usually contain words related to two or more different topics. While the Sense2Vec model can differentiate between words that can be used as a verb or as a noun, this process still generates one representation per POS tag. Figure 4 shows the most common words related to *troyano* (trojan in Spanish) in the Word2Vec model.DENISE classified this word as a valid candidate but, on further inspection, when selecting 300 most common words we can observe two clusters of words: one on the upper-left part that is related to its mythological sense and a small cluster on the lower-right that contains words related to the CS field.

Based on this observation, a possible future line of work might be the development of polysemic embeddings, be it as an added layer during the training process that could generate more than one representations for each word, or as a post training process using clustering or classification techniques. Such embeddings could be a good addition for other NLP related tasks such as automatic translation or Automatic Text Summarization (Torres-Moreno, 2014) or word sense disambiguation.

*Acknowledgements*

*Authors want to thank CONACYT (https://www.conacyt.gob.mx) Convocatoria de Becas al Extranjero 2015-2019, for supporting this research.*

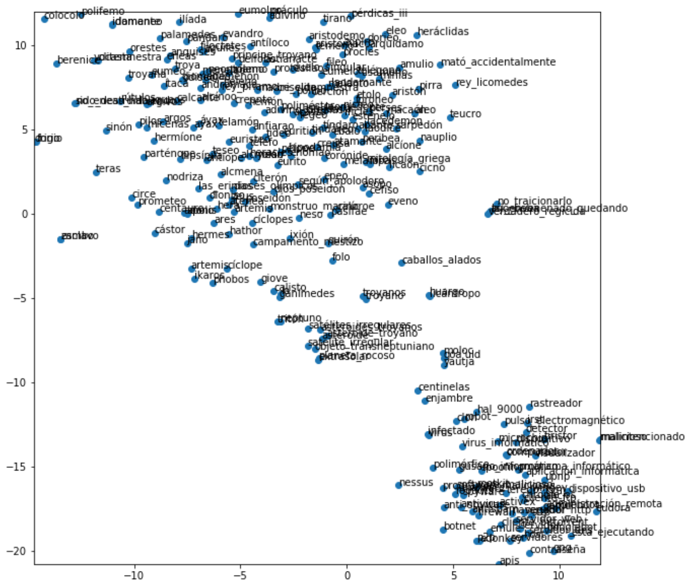


Figure 4: Most Common Words Related to “troyano” in Spanish

## References

Barrios F., López F., Argerich L., Wachenchauzer R. (2016, February). Variations of the Similarity Function of TextRank for Automated Summarization. *CoRR*, Vol. abs/1602.0. Retrieved from http://arxiv.org/abs/1602.03606

Bernhard D., Bruneau L., Falk I., Gérard C. (2015a, September). Création lexicale et corpus dynamique: quelles variables contextuelles? In *10e journées internationales ltt 2015 à strasbourg: La création lexicale en situation: texte, genres, cultures.* Strasbourg, France.

Bernhard D., Bruneau L., Falk I., Gérard C. (2015b, October). Le logoscope : une approche textuelle de la veille néologique. des raisons linguistiques à l’interface web. In *Cineo 2015 - iii congreso internacional de neología en las lenguas románicas.* Salamanca, Spain.

Bojanowski P., Grave E., Joulin A., Mikolov T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Cabré M. T. (2009). La classificació dels neologismes: una tasca complexa. In M. T. Cabré, R. Estopà (Eds.), *Les paraules noves: criteris per detectar i mesurar els neologismes*, pp. 11–37. Vic, Barcelona, Eumo Editorial, Universitat Pompeu Fabra.

Cabré M. T., Estopà R. (2005). Unidades de conocimiento especializado: caracterización y tipología. In M. T. Cabré, B. Carme (Eds.), *Coneixement, llenguatge i discurs especialitzat*, pp. 69–93. Barcelona, Institut Universitari de Lingüística Aplicada. Universitat Pompeu Fabra.

Estopà R. (2013). Les aplicacions terminològiques. In O. Domènech, R. Estopà (Eds.), *Llenguatges d’especialitat i terminologia.* Barcelona, Universitat Oberta de Catalunya.

Falk I., Bernhard D., Gérard C. (2014a, July). De la quenelle culinaire à la quenelle politique : identification de changements sémantiques à l'aide des Topic Models. In *21ème conférence sur le Traitement Automatique des Langues Naturelles.* Marseille, France.

Falk I., Bernhard D., Gérard C. (2014b, May). From Non Word to New Word: Automatically Identifying Neologisms in French Newspapers. In *LREC - The 9th edition of the Language Resources and Evaluation Conference.* Reykjavik, Iceland.

Falk I., Bernhard D., Gérard C., Potier-Ferry R. (2014, July). Étiquetage morpho-syntaxique pour des mots nouveaux. In *21ème conférence sur le Traitement Automatique des Langues Naturelles.* Marseille, France.

Gérard C., Falk I., Bernhard D. (2014, July). Traitement automatisé de la néologie : pourquoi et comment intégrer l'analyse thématique ? In *Actes du 4e Congrès Mondial de Linguistique Française (CMLF 2014)*, Vol. 8, p. 2627 - 2646. Berlin, Germany.

Janssen M. (2005). NeoTrack: semi-automatic neologism detection. In *Apl xxi.* Porto, Portugal.

Janssen M. (2009). Detección de Neologismos: una perspectiva computacional. *Debate Terminológico*, Vol. 5, No. 05, pp. 68–75.

Janssen M. (2012). NeoTag: a POS Tagger for Grammatical Neologism Detection. *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, No. 1, pp. 2118–2124.

Joulin A., Grave E., Bojanowski P., Mikolov T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759.*

Li G., Wang H. (2014). Improved Automatic Keyword Extraction Based on TextRank Using Domain Knowledge. In *Natural Language Processing and Chinese Computing*, pp. 403–413.

Mihalcea R., Tarau P. (2004). TextRank: Bringing Order into Text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing.*

Mikolov T., Chen K., Corrado G., Dean J. (2013). Efficient Estimation of Word Representations in Vector Space. *CoRR*, Vol. abs/1301.3.

Mikolov T., Sutskever I., Chen K., Corrado G., Dean J. (2013). Distributed representations of words and phrases and their compositionality. *CoRR*, Vol. abs/1310.4546.

Nazar R. (2011). *Neología semántica: un enfoque desde la lingüística cuantitativa.*

Nazar R. (2013). Word sense discrimination using statistic analysis of texts. *Barcelona Investigación Arte Creación*, Vol. 1, No. 1, pp. 5–26.

Nazar R. (2014). *Una metodología para depurar los resultados de los extractores de términos* (C. V. Sierra, Ed.). Comares.

Page L., Brin S., Motwani R., Winograd T. (1999, November). *The pagerank citation ranking: Bringing order to the web.*. Technical Report No. 1999-66. Stanford InfoLab.

Pay T., Lucci S., Cox J. (2018). An Ensemble of Automatic Keyphrase Extractors : TextRank , RAKE and TAKE. , No. August.

Renouf A. (1998). Explorations in corpus linguistics. In A. Renouf, R. H. Baayen (Eds.),, chap. Aviating among the Hapax Legomena: morphological grammaticalisation in current British Newspaper English. Rodopi.

Renouf A. (2010). Identification automatique de la néologie lexicologique et sémantique : questions soulevées par notre méthode. In M. T. Cabré, O. Domènech, R. Estopà, J. Freixa, M. Lorente (Eds.), *Actes del congrés internacional de neologia de les llengües romàniques*, pp. 129–141. Barcelona, Intitut Universitari de Lingüística Aplicada.

Renouf A. (2012). A finer definition of neology in English: the life-cycle of a word. *Corpus perspectives on patterns of lexis*, pp. 177–208.

Reutenauer C., Jacquey E., Ollinger S., De N., Romanes L. (2011). *Neologismes de sens: contribution à leur caracterisation dans un corpus autour du th àme de la crise financière.* Sao Paulo, Brasil.

Sablayrolles J.-F. (2012). Extraction automatique et types de néologismes : une nécessaire clarification. *Cahier de lexicologie*, Vol. 100, No. 1, pp. 37–53.

Tebé C. (2002). Bases pour une sélection de neologismes. In E. S. M. Teresa Cabré Judit Freixa (Ed.), *Lèxic i neologia*, pp. 43–50. Barcelona, Observatori de Neologia and Universitat Pompeu Fabra.

Torres-Moreno J.-M. (2014). *Automatic Text Summarization*. London : Hoboken, NJ, ISTE Ltd ; John Wiley & Sons, Inc.

Trask A., Michalak P., Liu J. (2015). sense2vec - a fast and accurate method for word sense disambiguation in neural word embeddings. *CoRR*, Vol. abs/1511.06388.