# ALL-IN-1 at IJCNLP-2017 Task 4:
# Short Text Classification with One Model for All Languages

**Barbara Plank**
Center for Language and Cognition
University of Groningen
`b.plank@rug.nl`

## Abstract

We present ALL-IN-1, a simple model for multilingual text classification that does not require any parallel data. It is based on a traditional Support Vector Machine classifier exploiting multilingual word embeddings and character n-grams. Our model is simple, easily extendable yet very effective, overall ranking 1st (out of 12 teams) in the IJCNLP 2017 shared task on customer feedback analysis in four languages: English, French, Japanese and Spanish.

## 1 Introduction

Customer feedback analysis is the task of classifying short text messages into a set of predefined labels (e.g., bug, request). It is an important step towards effective customer support.

However, a real bottleneck for successful classification of customer feedback in a multilingual environment is the limited transferability of such models, i.e., typically each time a new language is encountered a new model is built from scratch. This is clearly impractical, as maintaining separate models is cumbersome, besides the fact that existing annotations are simply not leveraged.

In this paper we present our submission to the IJCNLP 2017 shared task on customer feedback analysis, in which data from four languages was available (English, French, Japanese and Spanish). Our goal was to build a single system for all four languages, and compare it to the traditional approach of creating separate systems for each language. We hypothesize that a single system is beneficial, as it can provide positive transfer, particularly for the languages for which less data is available. The contributions of this paper are:

- We propose a very simple multilingual model for four languages that overall ranks first (out of 12 teams) in the IJCNLP 2017 shared task on Customer Feedback Analysis.

- We show that a traditional model outperforms neural approaches in this low-data scenario.

- We show the effectiveness of a very simple approach to induce multilingual embeddings that does not require any parallel data.

- Our ALL-IN-1 model is particularly effective on languages for which little data is available.

- Finally, we compare our approach to automatic translation, showing that translation negatively impacts classification accuracy.

- To support reproducibility and follow-up work all code is available at: https://github.com/bplank/ijcnlp2017-customer-feedback

## 2 ALL-IN-1: One Model for All

Motivated by the goal to evaluate how good a single model for multiple languages fares, we decided to build a very simple model that can handle any of the four languages. We aimed at an approach that does *not* require any language-specific processing (beyond tokenization) nor requires any parallel data. We set out to build a simple baseline, which turned out to be surprisingly effective. Our model is depicted in Figure 1.
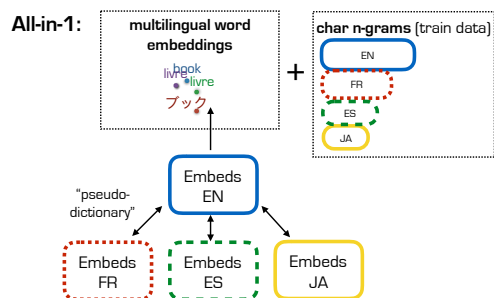


Figure 1: Overview of our ALL-IN-1 model.

Our key motivation is to provide a simple, general system as opposed to the usual ad-hoc setups one can expect in a multilingual shared task. So we rely on character n-grams, word embeddings, and a traditional classifier, motivated as follows.

First, character n-grams and traditional machine learning algorithms have proven successful for a variety of classification tasks, e.g., native language identification and language detection. In recent shared tasks simple traditional models outperformed deep neural approaches like CNNs or RNNs, e.g., (Medvedeva et al., 2017; Zampieri et al., 2017; Malmasi et al., 2017; Kulmizev et al., 2017). This motivated our choice of using a traditional model with character n-gram features.

Second, we build upon the recent success of multilingual embeddings. These are embedding spaces in which word types of different languages are embedded into the same high-dimensional space. Early approaches focus mainly on bilingual approaches, while recent research aims at mapping several languages into a single space. The body of literature is huge, but an excellent recent overview is given in Ruder (2017). We chose a very simple and recently proposed method that does not rely on any parallel data (Smith et al., 2017) and extend it to the multilingual case. In particular, the method falls under the broad umbrella of *monolingual mappings*. These approaches first train monolingual embeddings on large unlabeled corpora for the single languages. They then learn linear mappings between the monolingual embeddings to map them to the same space. The approach we apply here is particularly interesting as it does not require parallel data (parallel sentences/documents or dictionaries) and is readily applicable to off-the-shelf embeddings. In brief, the approach aims at learning a transformation in which word vector spaces are orthogonal (by applying SVD) and it leverages so-called "pseudo-dictionaries". That is, the method first finds the common word types in two embedding spaces, and uses those as pivots to learn to align the two spaces (cf. further details in Smith et al. (2017)).

## 3 Experimental Setup

In this section we first describe the IJCNLP 2017 shared task 4[1] including the data, the features, model and evaluation metrics.

---

|        | EN   | ES   | FR   | JP   |
|--------|------|------|------|------|
| TRAIN  | 3066 | 1632 | 1951 | 1527 |
| DEV    | 501  | 302  | 401  | 251  |
| TEST   | 501  | 300  | 401  | 301  |

Table 1: Overview of the dataset (instances).

### 3.1 Task Description

The customer feedback analysis task (Liu et al., 2017) is a short text classification task. Given a customer feedback message, the goal is to detect the type of customer feedback. For each message, the organizers provided one or more labels. To give a more concrete idea of the data, the following are examples of the English dataset:

- "Still calls keep dropping with the new update" (*bug*)

- "Room was grubby, mold on windows frames." (complaint)

- "The new update is amazing." (*comment*)

- "Needs more control s and tricks.." (*request*)

- "Enjoy the sunshine!!" (*meaningless*)

### 3.2 Data

The data stems from a joint ADAPT-Microsoft project. An overview of the provided dataset is given in Table 1. Notice that the available amount of data differs per language.
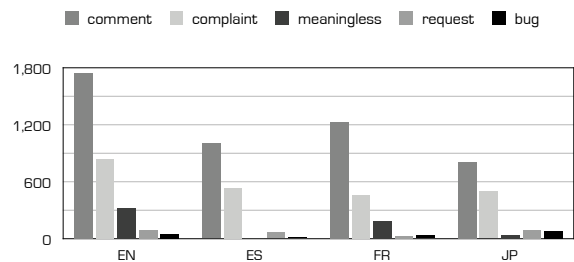


Figure 2: Distribution of the labels per language.

We treat the customer feedback analysis problem as a single-class classification task and actually ignore multi-label instances, as motivated next. The final label distribution for the data is given in Figure 2.

In initial investigations of the data we noticed that very few instances had multiple labels, e.g., "*comment,complaint*". In the English training data this amounted to ∼4% of the data. We decided to ignore those additional labels (just picked the first

in case of multiple labels) and treat the problem as a single-class classification problem. This was motivated by the fact that some labels were expected to be easily confused. Finally, there were some labels in the data that did not map to any of the labels in the task description (i.e., 'undetermined', 'undefined', 'nonsense' and 'noneless', they were presumably typos) so we mapped them all to the 'meaningless' label. This frames the task as a 5-class classification problem with the following classes:

- *bug*,
- *comment*,
- *complaint*,
- *meaningless* and
- *request*.

At test time the organizers additionally provided us with *translations* of the three language-specific test datasets back to English. These translations were obtained by Google translate. This allowed us to evaluate our English model on the translations, to gauge whether translation is a viable alternative to training a multilingual model.

## 3.3 Pre-processing

We perform two simple preprocessing steps. First of all, we tokenize all data using off-the-shelf tokenizers. We use `tinysegmenter`[2] for Japanese and the NLTK `TweetTokenizer` for all other languages. The Japanese segmenter was crucial to get sufficient coverage from the word embeddings later. No additional preprocessing is performed.

## 3.4 Multilingual Embeddings

Word embeddings for single languages are readily available, for example the Polyglot[3] or Facebook embeddings (Bojanowski et al., 2016), which were recently released.

In this work we start from the monolingual embeddings provided by the Polyglot project (Al-Rfou et al., 2013). We use the recently proposed approach based on SVD decomposition and a "pseudo-dictionary" (Smith et al., 2017) obtained from the monolingual embeddings to project embedding spaces. To extend their method from the

bilingual to the multilingual case, we apply pairwise projections by using English as pivot, similar in spirit to Ammar et al. (2016). We took English as our development language. We also experimented with using larger embeddings (Facebook embeddings; larger in the sense of both trained on more data and having higher dimensionality), however, results were comparable while training time increased, therefore we decided to stick to the smaller 64-dimensional Polyglot embeddings.

## 3.5 Model and Features

As classifier we use a traditional model, a Support Vector Machine (SVM) with linear kernel implemented in `scikit-learn` (Pedregosa et al., 2011). We tune the regularization parameter $C$ on the English development set and keep the parameter fixed for the remaining experiments and all languages ($C = 10$).

We compared the SVM to `fastText` (Joulin et al., 2016). As we had expected `fastText` gave consistently lower performance, presumably because of the small amounts of training data. Therefore we did not further explore neural approaches.

Our features are character n-grams (3-10 grams, with binary tf-idf) and word embeddings. For the latter we use a simple continuous bag-of-word representation (Collobert et al., 2011) based on averaging and min-max scaling.

Additionally, we experimented with adding Part-Of-Speech (POS) tags to our model. However, to keep in line with our goal to build a *single system for all languages* we trained a single multilingual POS tagger by exploiting the projected multilingual embeddings. In particular, we trained a state-of-the-art bidirectional LSTM tagger (Plank et al., 2016)[4] that uses both word and character representations on the concatenation of language-specific data provided from the Universal Dependencies data (version 1.2 for En, Fr and Es and version 2.0 data for Japanese, as the latter was not available in free-form in the earlier version). The word embeddings module of the tagger is initialized with the multilingual embeddings. We investigated POS n-grams (1 to 3 grams) as additional features.

---

[2]https://pypi.python.org/pypi/tinysegmenter

[3]Despite their name the Polyglot embeddings are actually monolingual embeddings, but available for many languages.

[4]https://github.com/bplank/bilstm-aux

|  | EN | ES | FR | JP | AVG |
|---|---|---|---|---|---|
| *MONOLINGUAL MODELS* | | | | | |
| Embeds | 50.6 | 82.0 | 66.5 | 65.1 | 66.05 |
| Words (W) | 66.1 | 86.9 | 73.2 | 73.6 | 74.95 |
| Chars (C) | 68.2 | 87.1 | 76.1 | 74.0 | 76.35 |
| W+Chars (C) | 65.9 | 87.7 | 75.7 | 74.0 | 75.82 |
| C+Embeds‡ | 66.1 | 86.6 | 76.5 | 77.1 | 76.58 |
| W+C+Embeds | 65.9 | 87.8 | 75.6 | 76.8 | 76.52 |
| *BILINGUAL MODEL* | | | | | |
| En+Es | 67.6 | 86.6 | – | – | – |
| En+Fr | 66.6 | – | 77.8 | – | – |
| En+Jp | 66.7 | – | – | 77.9 | – |
| *MULTILINGUAL MODEL* | | | | | |
| En+Es+Fr | 68.3 | 87.0 | 77.9 | – | – |
| ALL-IN-1‡ | 68.8 | 87.7 | 76.4 | 77.2 | 77.5 |
| ALL-IN-1+POS | 68.4 | 86.0 | 74.4 | 74.5 | 75.8 |

Table 2: Results on the development data, weighted F1. MONOLINGUAL: per-language model; MULTILINGUAL: ALL-IN-1 (with C+Embeds features trained on En+Es+Fr+Jp). ‡ indicates submitted systems.

## 3.6 Evaluation

We decided to evaluate our model using weighted F1-score, i.e., the per-class F1 score is calculated and averaged by weighting each label by its support. Notice, since our setup deviates from the shared task setup (single-label versus multi-label classification), the final evaluation metric is different. We will report on weighted F1-score for the development and test set (with simple macro averaging), but use Exact-Accuracy and Micro F1 over all labels when presenting official results on the test sets. The latter two metrics were part of the official evaluation metrics. For details we refer the reader to the shared task overview paper (Liu et al., 2017).

## 4 Results

We first present results on the provided development set, then on the official evaluation test set.

## 4.1 Results on Development

First of all, we evaluated different feature representations. As shown in Table 2 character n-grams alone prove very effective, outperforming word n-grams and word embeddings alone. Overall simple character n-grams (C) in isolation are often more beneficial than word and character n-grams together, albeit for some languages results

|  | EN | ES | FR | JP | AVG |
|---|---|---|---|---|---|
| MONOLING | **68.6** | 88.2 | **76.1** | 74.3 | 76.8 |
| MULTILING | 68.1 | **88.7** | 73.9 | **76.7** | **76.9** |
| TRANSLATE | – | 83.4 | 69.5 | 61.6 | – |

Table 3: Results on the test data, weighted F1. MONOLING: monolingual models. MULTILING: the multilingual ALL-IN-1 model. TRANS: translated targets to English and classified with EN model.

are close. The best representation are character n-grams with word embeddings. This representation provides the basis for our multilingual model which relies on multilingual embeddings. The two officially submitted models both use character n-grams (3-10) and word embeddings. Our first official submission, MONOLINGUAL is the per-language trained model using this representation.

Next we investigated adding more languages to the model, by relying on the multilingual embeddings as bridge. For instance in Table 2, the model indicated as En+Es is a character and word embedding-based SVM trained using bilingual embeddings created by mapping the two monolingual embeddings onto the same space and using both the English and Spanish training material. As the results show, using multiple languages can improve over the in-language development performance of the character+embedding model. However, the bilingual models are still only able to handle pairs of languages. We therefore mapped all embeddings to a common space and train a single multilingual ALL-IN-1 model on the union of all training data. This is the second model that we submitted to the shared task. As we can see from the development data, on average the multilingual model shows promising, overall (macro average) outperforming the single language-specific models. However, the multilingual model does not consistently fare better than single models, for example on French a monolingual model would be more beneficial.

Adding POS tags did not help (cf. Table 2), actually dropped performance. We disregard this feature for the final official runs.

## 4.2 Test Performance

We trained the final models on the concatenation of TRAIN and DEV data. The results on the test set (using our internally used weighted F1 metric)

are given in Table 3.

There are two take-away points from the main results: First, we see a positive transfer for languages with little data, i.e., the single multilingual model outperforms the language-specific models on the two languages (Spanish and Japanese) which have the least amount of training data. Overall results between the monolingual and multilingual model are close, but the advantage of our multilingual ALL-IN-1 approach is that it is a single model that can be applied to all four languages. Second, automatic translation harms, the performance of the EN model on the translated data is substantially lower than the respective in-language model. We could investigate this as the organizers provided us with translations of French, Spanish and Japanese back to English.

|  | EN | ES | FR | JP | AVG |
|---|---|---|---|---|---|
| Ours (MULTILING ) | 68.60 | **88.63** | 71.50 | **75.00** | **76.04** |
| Ours (MONOLING) | 68.80 | 88.29 | **73.75** | 73.33 | 75.93 |
| YNU-HPP-glove† | **71.00** | – | – | – | – |
| FYZU-bilstmcnn | 70.80 | – | – | – | – |
| IITP-CNN/RNN | 70.00 | 85.62 | 69.00 | 63.00 | 71.90 |
| TJ-single-CNN† | 67.40 | – | – | – |  |
| Baseline | 48.80 | 77.26 | 54.75 | 56.67 | 59.37 |

Table 4: Final test set results (Exact accuracy) for top 5 teams (ranked by macro average accuracy). Rankings for micro F1 are similar, we refer to the shared task paper for details. Winning system per language in bold. †: no system description available at the time of writing this description paper.

Averaged over all languages our system ranked first, cf. Table 4 for the results of the top 5 submissions. The multilingual model reaches the overall best exact accuracy, for two languages training a in-language model would be slightly more beneficial at the cost of maintaining a separate model. The similarity-based baseline provided by the organizers[5] is considerably lower.

Our system was outperformed on English by three teams, most of which focused only on English. Unfortunately at the time of writing there is no system description available for most other top systems, so that we cannot say whether they used more English-specific features. From the system names of other teams we may infer that most teams used neural approaches, and they score

|  | comm | compl | req | ml | bug |
|---|---|---|---|---|---|
| EN (MONOLING) | **82.3** | 64.4 | **60.0** | 27.5 | 0 |
| EN (MULTILING) | 82.0 | 65.0 | 42.1 | **28.6** | 0 |
| ES (MONOLING) | 93.3 | 75.2 | **72.7** | 0 | 0 |
| ES (MULTILING) | **93.5** | **76.2** | 66.6 | 0 | 66.6 |
| ES (TRANSLATE) | 92.6 | 67.2 | 11.8 | 0 | 0 |
| FR (MONOLING) | **86.4** | 65.6 | 14.3 | **47.6** | 54.5 |
| FR (MULTILING) | 85.5 | 61.5 | **16.6** | 41.2 | 50.0 |
| FR (TRANSLATE) | 82.9 | 58.9 | 16.6 | 34.5 | 0 |
| JP (MONOLING) | 85.7 | **67.8** | 55.8 | 0 | 50.0 |
| JP (MULTILING ) | **87.0** | 67.8 | 65.2 | 0 | 50.0 |
| JP (TRANSLATE ) | 76.5 | 61.3 | 7.2 | 0 | 0 |

Table 5: Test set results (F1) per category (*comment (comm)*, *complaint (compl)*, *request (req)*, *meaningless (ml)* and *bug*), official evaluation.

worse than our SVM-based system.

The per-label breakdown of our systems on the official test data (using micro F1 as calculated by the organizers) is given in Table 5. Unsurprisingly less frequent labels are more difficult to predict.

## 5 Conclusions

We presented a simple model that can effectively handle multiple languages in a single system. The model is based on a traditional SVM, character n-grams and multilingual embeddings. The model ranked first in the shared task of customer feedback analysis, outperforming other approaches that mostly relied on deep neural networks.

There are two take-away messages of this work: 1) multilingual embeddings are very promising[6] to build single multilingual models; and 2) it is important to compare deep learning methods to simple traditional baselines; while deep approaches are undoubtedly very attractive (and fun!), we always deem it important to compare deep neural to traditional approaches, as the latter often turn out to be surprisingly effective. Doing so will add to the literature and help to shed more light on understanding why and when this is the case.

## Acknowledgments

# References

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed Word Representations for Multilingual NLP. In *CoNLL*.

Waleed Ammar, George Mulcaire, Yulia Tsvetkov, Guillaume Lample, Chris Dyer, and Noah A Smith. 2016. Massively Multilingual Word Embeddings. *arXiv preprint arXiv:1602.01925*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Artur Kulmizev, Bo Blankers, Johannes Bjerva, Malvina Nissim, Gertjan van Noord, Barbara Plank, and Martijn Wieling. 2017. The power of character n-grams in native language identification. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, Copenhagen, Denmark. Association for Computational Linguistics.

Chao-Hong Liu, Yasufumi Moriya, Alberto Poncelas, Declan Groves, Akira Hayakawa, and Qun Liu. 2017. Introduction to ijcnlp 2017 shared task on customer feedback analysis. In *Proceedings of the 8th International Joint Conference on Natural Language Processing*, pages 1–7, Taipei, Taiwan. Association for Computational Linguistics.

Shervin Malmasi, Keelan Evanini, Aoife Cahill, Joel Tetreault, Robert Pugh, Christopher Hamill, Diane Napolitano, and Yao Qian. 2017. A report on the 2017 native language identification shared task. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, Copenhagen, Denmark. Association for Computational Linguistics.

Maria Medvedeva, Martin Kroon, and Barbara Plank. 2017. When sparse traditional models outperform dense neural networks: the curious case of discriminating between similar languages. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, Valencia, Spain. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 412–418, Berlin, Germany. Association for Computational Linguistics.

Sebastian Ruder. 2017. A survey of cross-lingual embedding models. *CoRR*, abs/1706.04902.

Samuel L Smith, David HP Turban, Steven Hamblin, and Nils Y Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *arXiv preprint arXiv:1702.03859*.

Marcos Zampieri, Shervin Malmasi, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, Jörg Tiedemann, Yves Scherrer, and Noëmi Aepli. 2017. Findings of the vardial evaluation campaign 2017. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, Valencia, Spain. Association for Computational Linguistics.