Multilingual Text Classification Made Easy

Alejandro Moreo Fernández

(Joint work with Andrea Esuli and Fabrizio Sebastiani)

Istituto di Scienza e Tecnologie dell'Informazione
Consiglio Nazionale delle Ricerche
56124 Pisa, IT
E-mail: alejandro.moreo@isti.cnr.it

IIR 2018, Roma, Italy
May 28–30, 2018

What is this talk about?

- Multilingual text classification
- Classifier ensembles
- Vector spaces

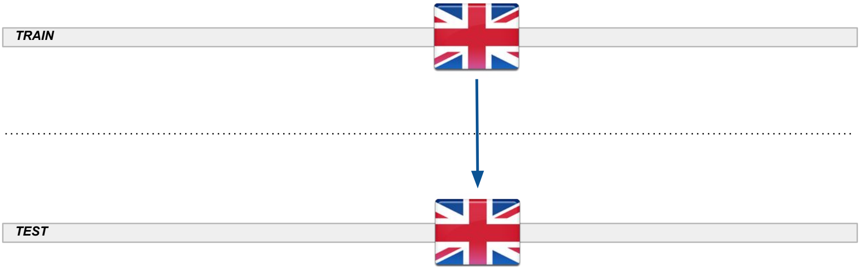
Text Classification

TRAIN

TEST

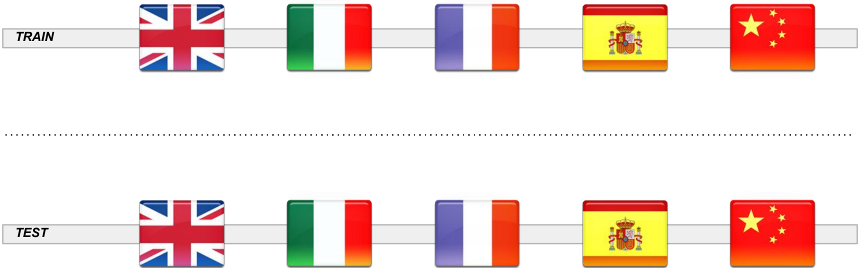
- Classification scheme ("codeframe") $\mathcal{C} = \{c_1, ..., c_n\}$

Text Classification



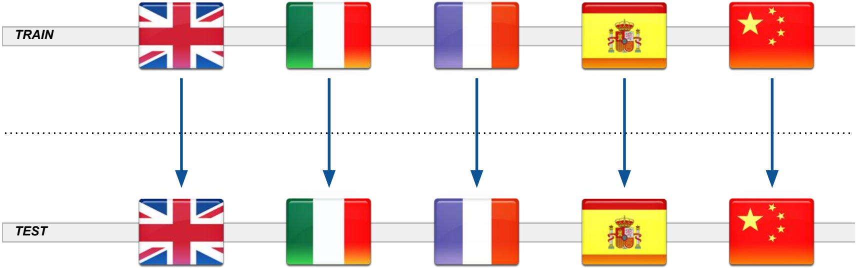
- Classification scheme ("codeframe") $\mathcal{C} = \{c_1, ..., c_n\}$
- We learn, by observing labelled (English) documents, a classifier (e.g., a SVM) for unlabelled (English) documents.

Multilingual Text Classification



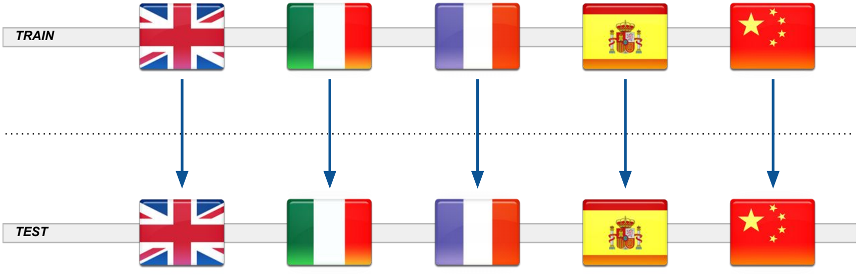
- Each document d written in one of a finite set $\mathcal{L} = \{\lambda_1, , ..., \lambda_m\}$
- Classification scheme ("codeframe") $\mathcal{C} = \{c_1, ..., c_n\}$ is the same for all languages
- Scenario common in many multinational organizations (e.g., European Union) / companies (e.g., Vodafone)
- How can we learn from heterogeneous data?

The Naïve Solution



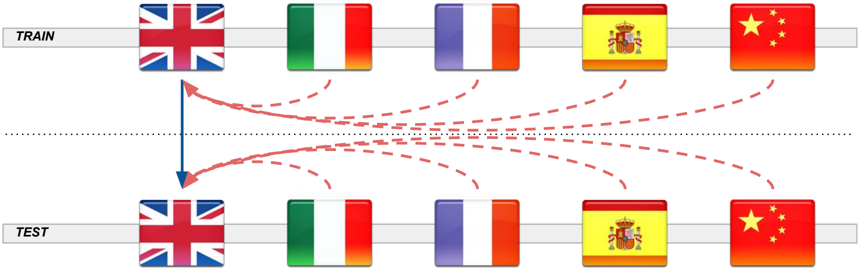
- MLC solved as m independent monolingual classification tasks

The Naïve Solution



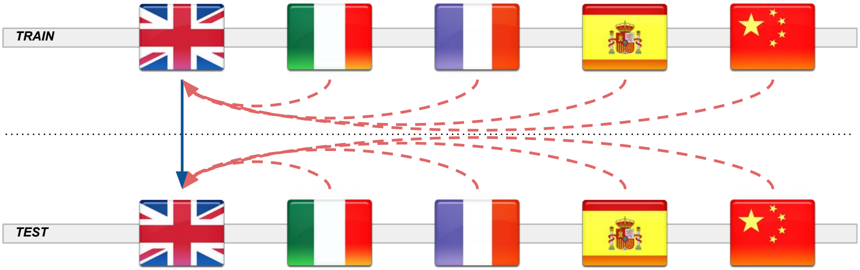
- MLC solved as m independent monolingual classification tasks
- Suboptimal!

The Machine Translation approach



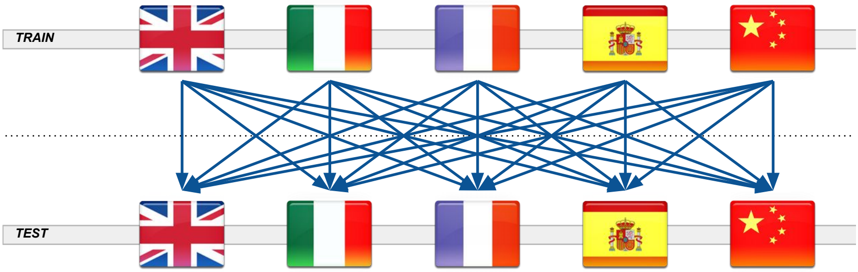
- Use MT to transform all documents into a single language.

The Machine Translation approach



- Use MT to transform all documents into a single language.
- Problems:
  - MT tools may not be available for certain language pairs,
  - may not be free
  - may work suboptimally

Poly-lingual Text Classification



- Attempts to exploit synergies among languages
- $\Rightarrow$ Improve on monolingual classifiers (naïve)

Poly-lingual Text Classification

- And we want to avoid the use of any:

Poly-lingual Text Classification

- And we want to avoid the use of any:
  - MT tools

Poly-lingual Text Classification

- And we want to avoid the use of any:
  - MT tools
  - Bi-lingual dictionaries

Poly-lingual Text Classification

- And we want to avoid the use of any:
  - MT tools
  - Bi-lingual dictionaries
  - Multilingual Thesaurus (e.g., BabelNet)

Poly-lingual Text Classification

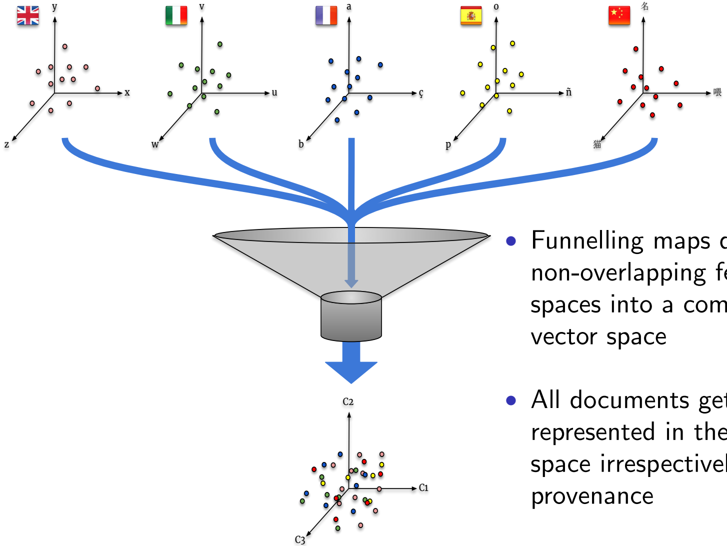
- And we want to avoid the use of any:
  - MT tools
  - Bi-lingual dictionaries
  - Multilingual Thesaurus (e.g., BabelNet)
  - External resources (e.g., Wikipedia)

Poly-lingual Text Classification

- And we want to avoid the use of any:
  - MT tools
  - Bi-lingual dictionaries
  - Multilingual Thesaurus (e.g., BabelNet)
  - External resources (e.g., Wikipedia)

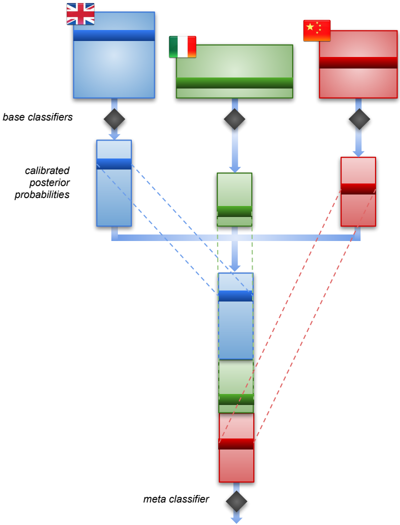
- Is that possible?

Funnelling!



- Funnelling maps different non-overlapping feature spaces into a common vector space
- All documents get represented in the common space irrespectively of their provenance

Funnelling: PLC made easy

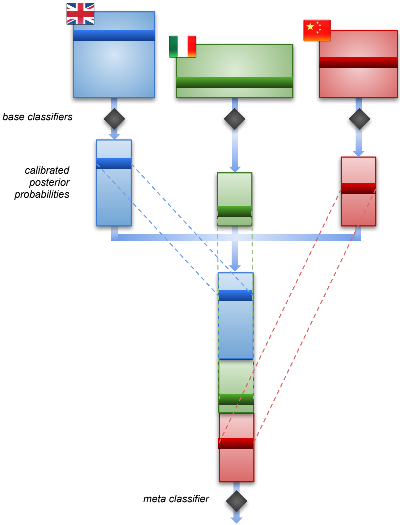


- Two-level classification architecture
  1. $|\mathcal{L}|$ language-dependent base classifiers
  2. One language-independent metaclassifier

- For the metaclassifier, document d represented as vector of $|\mathcal{C}|$ classification scores

- Metaclassifier outputs a vector of $|\mathcal{C}|$ classification scores

Funnelling: PLC made easy



- All documents from any language contribute to the other languages
- Learner-independent
- Independent from representation model used in base classifiers
- No requirement that training set should be parallel or comparable
- No requirement for ML dictionaries, ML datasets, MT services

Training a funnelling system

Fun(TAT): "Funnelling Training and Test"

- Train base classifiers using monolingual training sets
- Classify training examples via trained classifiers
- Uses classification scores of training examples for training metaclassifiers

Training a funnelling system

Fun(TAT): "Funnelling Training and Test"

- Train base classifiers using monolingual training sets
- Classify training examples via trained classifiers
- Uses classification scores of training examples for training metaclassifiers
    - Problem: base classifiers generate higher-quality representations for training data than for test data (iid assumption)

Training a funnelling system

Fun(TAT): "Funnelling Training and Test"

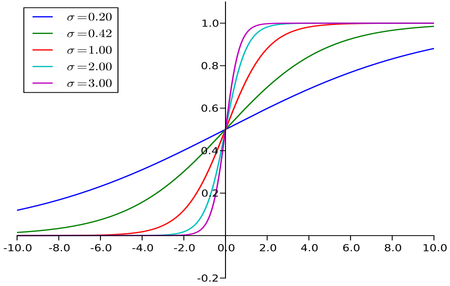
- Train base classifiers using monolingual training sets
- Classify training examples via trained classifiers
- Uses classification scores of training examples for training metaclassifiers
  - Problem: base classifiers generate higher-quality representations for training data than for test data (iid assumption)

Fun(kFCV): "Funnelling k-Fold Cross-Validation"

1. Train base classifiers using monolingual training sets (same)
2. Classify training examples via k-fold cross-validation
3. Use classification scores of training examples for training (same) metaclassifiers

Probability calibration

- Problem: metaclassifier receives as input vectors coming from different, incomparable sources
- Solution: make them comparable!, by converting classification scores $S(c, d)$ into well calibrated posterior probabilities $\Pr(c|d)$
- Calibration: "90% of items whose $\Pr(c|d)$ is 0.9 should belong to c"
- To be performed independently for each base classifier



Training a funnelling system: Fun(TAT)

Fun(TAT) :

1. Train base classifiers using monolingual training sets
2. Classify training examples via trained classifiers
3. Map classification scores into well-calibrated posterior probabilities
4. Use posterior probabilities of training examples for training metaclassifiers

Fun(kFCV) :

1. Train base classifiers using monolingual training sets
2. Classify training examples via k-fold cross-validation
3. Map classification scores into well-calibrated posterior probabilities
4. Use posterior probabilities of training examples for training metaclassifiers

How well does funnelling work?

Datasets and learners

- Datasets:
  - RCV1/RCV2: comparable corpus, 9 languages, 10 samples $\times$ ((1000 training + 1000 test) per language), 73 classes
  - JRC-Acquis: parallel corpus, 11 languages, 10 samples $\times$ ((1155 training + 4242 test) per language), 300 classes
- Learners:
  - SVMs w/ linear kernel (base classifiers)
  - SVMs w/ RBF kernel (metaclassifier)

Baselines and evaluation measures

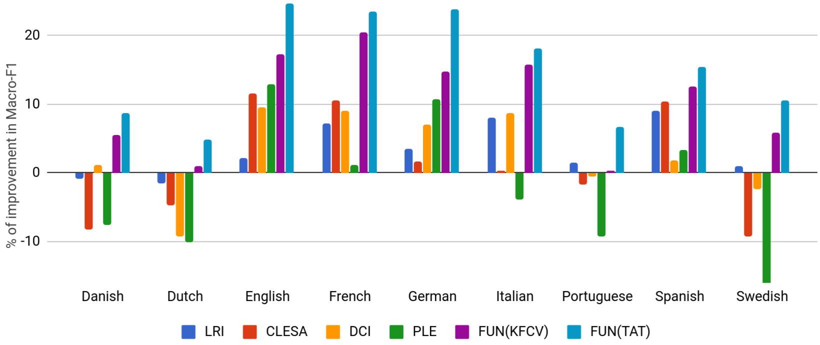
- Baselines:
  - Naïve (i.e., monolingual classification)
  - Cross-Lingual Explicit Semantic Analysis (CLESA – Song & Cimiano, CLEF 2008)
  - Distributional Correspondence Indexing (DCI – Moreo et al., JAIR 2016a)
  - Lightweight Random Indexing (LRI – Moreo et al., JAIR 2016b)
  - Polylingual Embeddings (PLE – Conneau et al., ICLR 2018)
- Measures (both in micro- and macro-averaged versions):
  - F_1
  - K ($\approx$ "balanced accuracy")

Multi-label PLC results

		NAÏVE	LRI	CLESA	DCI	PLE	FUN(KFCV)	FUN(TAT)	UPPERBOUND
F_1^{μ}	RCV1/RCV2	.776	.771	.714	.770	.696	.801†	**.802**	–
	JRC-Acquis	.559	**.594**	.557	.510	.478	.581	.587	.707
F_1^{M}	RCV1/RCV2	.467	.490	.471	.485	.453	.512	**.534**	–
	JRC-Acquis	.340	**.411**	.379	.317	.300	.356	.399	.599
K^{μ}	RCV1/RCV2	.690	.696	.659	.696	.644	.731	**.760**	–
	JRC-Acquis	.429	.476	.453	.382	.429	.457	**.490**	.632
K^{M}	RCV1/RCV2	.417	.440	.434	.456	.466	.482	**.506**	–
	JRC-Acquis	.288	.348	.330	.274	.349††	.328	**.365**	.547

Some results

- More consistent improvements over naïve baseline



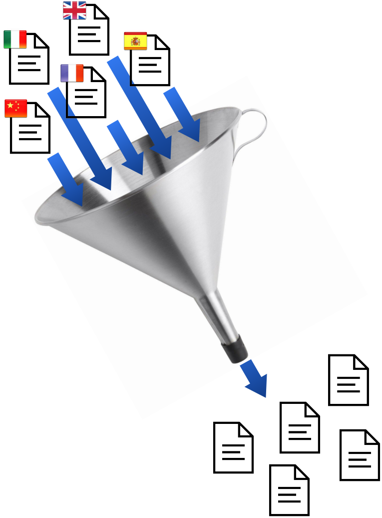
How efficient is funnelling?

	NAÏVE	LRI	CLESA	DCI	PLE	FUN(KFCV)	FUN(TAT)
RCV1/RCV2	537	5,506	28,508	344	954	1,041	**215**
	12	138	576	**3**	59	15	12
JRC-Acquis	6,005	67,571	63,497	4,888	**2,232**	13,127	4,987
	39	529	719	**8**	870	54	45

Conclusions

- PLC: an important task for many multinational organizations / companies
- Approach: mapping different language-dependent feature spaces into a language-independent vector space:
  - exploiting the information from all languages

Conclusions



- PLC: an important task for many multinational organizations / companies
- Approach: mapping different language-dependent feature spaces into a language-independent vector space:
  - exploiting the information from all languages
  - very effectively

Conclusions

- PLC: an important task for many multinational organizations / companies
- Approach: mapping different language-dependent feature spaces into a language-independent vector space:
  - exploiting the information from all languages
  - very effectively
  - very efficiently

Conclusions

- PLC: an important task for many multinational organizations / companies
- Approach: mapping different language-dependent feature spaces into a language-independent vector space:
  - exploiting the information from all languages
  - very effectively
  - very efficiently
  - using no external knowledge!

Where can we go from here?

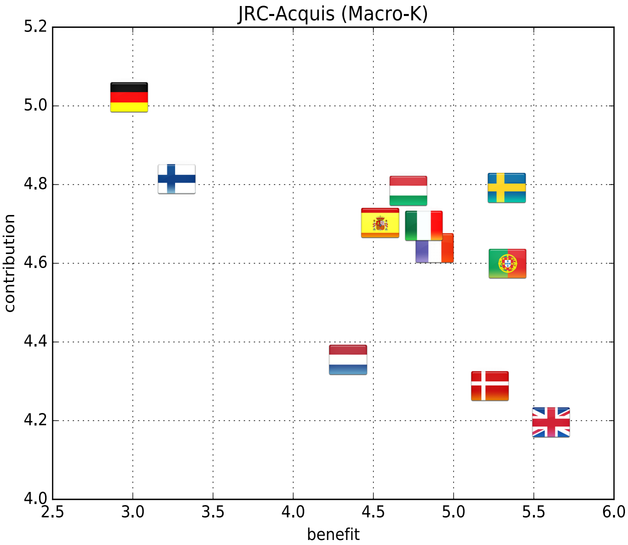
- Different codeframes
- Other classification scenarios (e.g., "multimodal" classification)
- Adopt a deep learning end-to-end architecture

Questions?

Thank you!

For any question, email me at
alejandro.moreo@isti.cnr.it

Which languages benefit / contribute most?



How does this contribution evolve?