Apprentissage par transfert et données mixtes pour évaluer l'importance de crues à partir d'articles d'information

Pierrick Bruneau, Yoann Didry, Thomas Tamisier

LIST, L-4362 Esch-sur-Alzette
pierrick.bruneau@list.lu

Résumé. Dans cet article applicatif, nous décrivons l'utilisation de propriétés textuelles et visuelles par des réseaux de neurones profonds pour évaluer l'importance des crues dans les articles d'information. En particulier, nous estimons la pertinence du transfert de modèles pré-entrainés sur des corpora conceptuellement proches. Nous évaluons également l'apport de modèles à branche double, qui combinent les représentations denses d'un texte et d'une image associée. Nous comparons la performance de ces variantes méthodologiques au moyen des données distribuées dans le cadre de l'atelier MediaEval MultiMedia Satellite (MMSat) 2019. Les résultats présentés ici ont fait l'objet d'une communication à l'atelier : le présent article propose une version significativement étendue des notes techniques accompagnant les prédictions réalisées sur l'ensemble de test de l'atelier MMSat.

1 Introduction

L'identification d'actualités en ligne liées à un événement catastrophique tel qu'une crue, et l'évaluation de l'étendue de cet événement grâce à l'information collectée peuvent être d'une importance critique en vue du secours aux victimes. Les tâches de désambiguïsation de sujet à partir de l'image (*News Image Topic Disambiguation* - NITD) et d'estimation multimodale du niveau de crue (*Multimodal Flood Level Estimation* - MFLE) de l'atelier MediaEval MultiMedia Satellite 2019 (Bischke et al., 2019) encouragent l'application de méthodologies d'apprentissage automatique dans ce contexte. Nous n'avons délibérément pas utilisé de modèle spécialisé à ces tâches, e.g. nous n'avons pas eu recours à des techniques de détection de pose ou d'occlusion (Bulat et Tzimiropoulos, 2016) pour la tâche MFLE.

Dans cette optique, la tâche NITD est un problème de classification d'images, qui peut être traité grâce un réseau de neurones convolutif tel que VGG16 (Simonyan et Zisserman, 2014), InceptionV3 (Szegedy et al., 2016), ou MobileNetV2 (Sandler et al., 2018). Ces modèles définissent de nombreux paramètres (souvent de l'ordre de plusieurs millions) qui sont coûteux à estimer à partir d'une initialisation aléatoire, et *a priori* difficilement sans un échantillon d'apprentissage conséquent. Une solution courante est alors l'apprentissage par transfert, qui consiste à réutiliser des modèles pré-entrainés sur une tâche voisine. De tels modèles pré-entrainés sont même souvent mis à disposition par des librairies d'apprentissage profond telles

que Keras [1]. Notre travail contribue à estimer la valeur de ce transfert dans le contexte de tâches spécifiques telles que NITD et MFLE.

Tels qu'employés par certains auteurs, les réseaux de neurones multimodaux visent à apprendre la similarité entre plusieurs modalités de données comme le texte et l'image, dans le but de sous-titrer les images automatiquement, par exemple (Kiros et al., 2014). En bref, une instance de chaque modalité est présentée à sa couche d'entrée respective. Après diffusion, une représentation vectorielle dense est obtenue pour chaque modalité. Une fonction de perte contrastive (Hadsell et al., 2006) est alors utilisée afin d'ajuster les représentations denses selon les associations fournies en tant que supervision. Dans notre travail, la multimodalité est vue comme l'usage conjoint de plusieurs modalités (i.e. texte et image) afin d'améliorer la prédiction d'une classe. En d'autres termes, au lieu de contraster les représentations denses de chaque modalité, nous fusionnons ces dernières en vue d'optimiser une fonction d'entropie croisée typique des problèmes de classification (Lopez-Fuentes et al., 2017; Audebert et al., 2019). Nos propositions pour la tâche MFLE visent notamment à évaluer l'amélioration apportée par de telles approches multimodales.

Après une rapide présentation des tâches de l'atelier MMSat, ainsi que des données associées mises à disposition, nous décrivons les principes généraux ayant guidé nos propositions, puis donnons des détails spécifiques à chaque modalité. Notre contribution consistant principalement à l'application adaptée de modèles de la littérature, ces sections détaillées associent état de l'art et réflexions spécifiques à l'atelier MMSat. Enfin, nous dévoilons la performance obtenue sur les ensembles de test par nos propositions, et commentons ces résultats.

2 L'atelier MMSat

Pour la tâche NITD, les participants à MMSat reçoivent un échantillon d'apprentissage de 5180 images, avec lesquelles ils doivent construire un classifieur binaire prédisant si l'article d'information dont l'image a été extraite traite d'une crue ou non. Tous les articles contiennent un mot-clé lié aux crues (e.g. *flood*), mais ils n'en traitent pas forcément. Ainsi, seul $\sim 10.1\%$ de l'échantillon d'apprentissage est issu d'articles traitant effectivement de crues. L'ensemble de test contient 1296 images.

Toutes les 4932 images de l'échantillon d'apprentissage de la tâche MFLE montrent des scènes de crue. Les participants à la tâche doivent construire un classifieur binaire prédisant si au moins une personne dans l'image a de l'eau jusqu'au-delà des genoux, donnant un aperçu de la sévérité de la crue. L'ensemble d'apprentissage est multimodal, car le texte de l'article est associé à chaque image. Cette tâche de classification est fortement déséquilibrée, car la classe positive est représentée par seulement $\sim 3.2\%$ de l'ensemble d'apprentissage.

Pour chaque tâche, les participants ont pu soumettre 5 ensembles de prédictions de test, à chaque fois issues d'architectures de modèles différentes dans notre cas. Selon la tâche et l'ensemble, seule l'information visuelle et/ou textuelle fournie devait être utilisée, ou tout ensemble de données tierces pouvait également être employé. Après cette phase initiale, les organisateurs ont retourné la performance en termes du score F1 (i.e. moyenne harmonique de la précision et du rappel) de la classe positive pour chaque ensemble de prédictions soumis. Les résultats présentés en section 4 ont fait l'objet d'une communication résumée à l'atelier MMSat (Bru-

1. `https://keras.io/applications/`

neau et Tamisier, 2019). Les notes techniques de l'atelier (Bischke et al., 2019) peuvent être consultées pour plus de détails sur la collecte et l'annotation des échantillons, ainsi que sur le contexte des tâches proposées [2].

3 Approche proposée

Tous les modèles présentés par la suite sont des réseaux de neurones entrainés sur 50 passes avec ordre de présentation des données aléatoire (sauf MobileNetV2, qui a été entrainé sur 80 passes). Une métrique de validation a été calculée à la fin de chaque passe, et le modèle qui maximise cette métrique a été finalement retenu. Comme les deux tâches sont significativement déséquilibrées, nous avons utilisé la métrique F1 sur la classe positive, qui s'avère être également la métrique utilisée en test par les organisateurs de l'atelier MMSat. Nous avons utilisé l'algorithme d'optimisation Adam (Kingma et Ba, 2015) avec ses paramètres par défaut, sauf dans les phases d'ajustement fin où un taux d'apprentissage de 10^{-4} a été utilisé (au lieu de 10^{-3}). Des *minibatches* de 32 éléments ont été utilisés, afin de prendre en compte la taille modeste des ensembles d'apprentissage, tout en gardant des chances raisonnables d'avoir au moins une instance positive dans chaque *minibatch*.

Pour chaque architecture de modèle testée, nous avons réalisé une validation croisée stratifiée à 5 tours, revenant ainsi à entrainer 5 modèles en retenant 20% de données de validation différentes pour chaque d'entre eux. Nous avons construit des ensembles à partir des modèles sélectionnés à chaque tour. Nous avons combiné les modèles selon la moyenne de leurs scores (i.e. en moyennant les sorties des modèles juste avant l'activation finale, et en calculant l'activation sur cette moyenne).

La manière canonique de gérer le déséquilibre des classes avec les réseaux de neurones est la pondération des instances. Cela revient à pondérer les termes positifs (resp. négatifs) de la fonction de perte par $1/P$ (resp. $1/N$). L'importance relative des instances de la classe majoritaire est ainsi diminuée. Pour la construction de nos ensembles de prédictions, nous avons considéré l'emploi de modèles pré-entrainés comme un recours à une information tierce. Par conséquent, tous les ensembles pour lesquels ce recours était exclu ont été obtenus à partir de modèles à initialisation aléatoire.

3.1 Information textuelle

Il est possible de catégoriser un contenu textuel en le traitant comme une séquence de mots utilisée en entrée d'un réseau de neurones récurrent tel que le modèle LSTM (Graves, 2012). Avant de traiter la tâche MFLE, nous avons comparé la performance de plusieurs architectures de réseaux de neurones dans le contexte de la classification textuelle : le LSTM et sa variante bidirectionnelle (BiLSTM) (e.g. utilisée par (Limsopatham et Collier, 2016) pour la reconnaissance d'entités nommées), le BiLSTM à attention (Zhou et al., 2016), le modèle à attention hiérarchique (Yang et al., 2016), le modèle à attention à tête multiple (Vaswani et al., 2017), et le modèle convolutif au niveau caractère (Zhang et al., 2015). Pour les comparer, nous avons utilisé des hyperparamètres similaires (*minibatches* de taille 64, représentations denses de taille 64, et vecteurs d'état de taille 128), ainsi que des tâches de classification de

2. Les données associées aux tâches sont accessibles à `https://github.com/multimediaeval/2019-Multimedia-Satellite-Task`

NITD				
Run 1	Run 2	Run 3	Run 4	Run 5
(MNV2)	*(InceptionV3)*		*(VGG16)*	
∅	*ImageNet*	*+ fine tuning*	*Places365*	*+ fine tuning*
85.1	81.0	79.6	89.0	**89.6**
MFLE				
Run 1	Run 2	Run 3	Run 4	Run 5
(MVN2)	*(LSTM)*	*(MNV2 & LSTM)*	*(IV3 & LSTM)*	*(VGG16 & LSTM)*
∅	∅	∅	*ImageNet*	*Places365*
56.6	56.5	57.6	**67.1**	66.0

TAB. 1. Scores F1 (%) sur les données de test NITD et MFLE. Nous indiquons les architectures utilisées, ainsi que les données utilisées pour leur entrainement. ∅ indique une initialisation aléatoire.

référence pertinentes en l'espèce : la prédiction de polarité des critiques cinématographiques IMDB [3], et la classification d'articles d'information AG [4]. Les ensembles d'apprentissage respectifs sont de taille 25000 et 120000. Chaque texte a été retaillé à 500 mots (en coupant ou ajoutant des mots vides selon la situation). Pour les modèles utilisant une couche de représentation dense des mots en entrée, nous avons comparé l'initialisation aléatoire à l'utilisation de vecteurs Glove pré-entrainés (Pennington et al., 2014). Finalement, sur les 2 corpora, aucune option n'a obtenu de performances significativement meilleures que le LSTM de base avec initialisation aléatoire. Nous supposons que pour de telles tâches de classification textuelle, un modèle monodirectionnel est suffisant pour capturer l'information utile, et que pour des textes raisonnablement courts, les mécanismes d'attention n'expriment pas leur potentiel. Ainsi, seul le LSTM est utilisé pour la tâche MFLE. Ce choix est compatible avec les contraintes imposées pour le modèle 2 (voir table 1), car aucune information extérieure n'est alors utilisée.

Pour la construction de ce modèle, les hyperparamètres ont été déterminés via une recherche par quadrillage. Nous avons finalement retenu des textes retaillés à 50 mots, des vecteurs d'état de taille 100, et des représentations de mots denses de taille 32.

3.2 Information visuelle

Pour la classification d'images dans les deux tâches, nous nous sommes concentrés sur 3 architectures éprouvées : InceptionV3 (Szegedy et al., 2016), MobileNetV2 (Sandler et al., 2018) et VGG16 (Simonyan et Zisserman, 2014). InceptionV3 a été utilisé en tant que composant dans de nombreuses contributuons récentes (e.g. (Lopez-Fuentes et al., 2017; Poplin et al., 2018; Chen et al., 2018)). Des versions pré-entrainées sur les données de la compétition ImageNet (Deng et al., 2009) sont facilement accessibles. En comparaison, MobileNetV2 a un faible nombre de paramètres ($\sim$ 2M, quand InceptionV3 en comporte $\sim$ 20M, et VGG16 $\sim$ 130M). VGG16 est également répandu, et existe pré-entrainé sur les données Places365 (Zhou

3. https://www.kaggle.com/iarunava/imdb-movie-reviews-dataset
4. http://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html

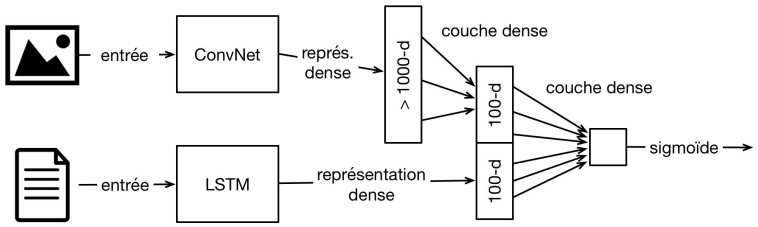

FIG. 1. Architecture à données mixtes générique.

et al., 2018) qui se focalisent sur la reconnaissance des lieux, ce qui semble *a priori* plus proche des tâches considérées ici.

Pour tous les modèles, les images ont été redimensionnées à 224×224 pixels. Nous avons appliqué les techniques d'augmentation de données communément employées dans les articles suscités, i.e. chaque image présentée à l'entrainement subit une combinaison de transformation aléatoires (décalages horizontaux et verticaux, cisaillement, zoom, miroir et décalage de luminosité). Les spécifications par défaut des couches neuronales ont été utilisées par ailleurs. Quand aucune information externe ne devait être utilisée (i.e. modèle 1 de NITD et MFLE), nous avons utilisé MobileNetV2 initialisé aléatoirement pour minimiser le nombre de paramètres à entrainer. Pour l'apprentissage par transfert, nous avons utilisé InceptionV3 (resp. VGG16) pré-entrainé avec ImageNet (resp. Places365). Pour InceptionV3, dans un premier temps, seule la dernière couche dense a été entrainée (modèle NITD 2). Les deux dernières couches convolutives du modèle résultant ont alors subi un ajustement fin (modèle NITD 3). Les 3 dernières couches de VGG16 sont denses, l'antépénultième contenant 75% du nombre de paramètres total du modèle. L'ajustement fin des dernières couches convolutives de ce modèle impose d'ajuster cette couche prépondérante avec un échantillon de taille comparativement faible. Pour contourner ce problème, nous avons directement entrainé les deux dernières couches denses (modèle NITD 4). Nous avons également proposé l'ajustement fin de l'antépénultième couche et de la dernière couche convolutive malgré tout (modèle NITD 5).

3.3 Données mixtes

La figure 1 montre l'architecture générique utilisée pour nos modèles à données mixtes. Notre approche est assez similaire à celle proposée par (Lopez-Fuentes et al., 2017). Pour le modèle MFLE 3, nous avons réutilisé le LSTM du modèle MFLE 2 et le MobileNetV2 du modèle MFLE 1, et n'avons entrainé que les couches denses connectés à leur suite. De manière analogue, le modèle MFLE 4 a mobilisé un InceptionV3 entrainé et ajusté finement sur les images de la tâche en suivant le protocole décrit en section 3.2. Un VGG16 ayant suivi le même processus a été utilisé pour le modèle 5. Des ensembles de 5 modèles multimodaux ont été entrainés à partir de cette base commune. Pour équilibrer l'influence du texte et de l'image, nous avons défini un goulot d'étranglement linéaire, qui réduit la représentation convolutive dense (e.g. 2048 for InceptionV3) à 100, la même taille que le vecteur d'état LSTM.

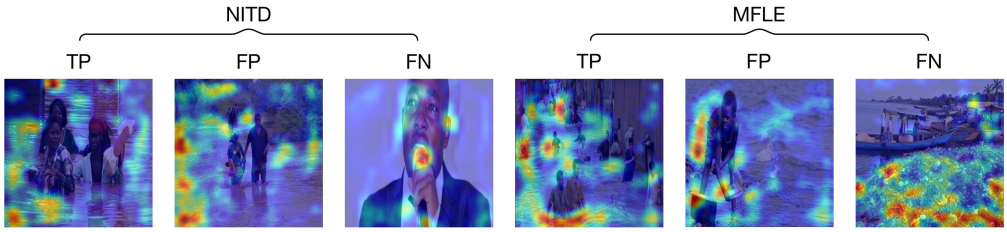

FIG. 2. *Class Activation Maps* pour des exemples de vrais (TP) et faux positifs (FP), et de faux négatifs (FN) dans les 2 tâches.

4 Analyse des résultats

Les scores F1 résultant de nos soumissions sont indiqués en table 1. Pour la tâche NITD, VGG16 après ajustement fin obtient le meilleur score (F1 = 89.6%). Les 2 modèles VGG16 en général sont significativement meilleurs. Cela suggère que NITD est proche d'une tâche de reconnaissance de scènes. Toutefois, l'ajustement fin apporte un gain de performance au mieux modeste (+0.6% pour VGG16, runs 4 et 5), au pire une dégradation (-1.4% pour InceptionsV3, runs 2 et 3). De manière surprenante, MobileNetV2 initialisé aléatoirement offre des performances solides, entre InceptionV3 et VGG16. Pour la tâche MFLE, le modèle à données mixtes bénéficie d'un léger effet synergique entre modalités (+1.0% entre les runs 1 et 3). L'utilisation de modèles visuels pré-entrainés (runs 4 et 5) engendre un gain de performance significatif ($\approx$ 10%). Les meilleurs résultats sont obtenus par la combinaison des modèles LSTM et InceptionV3 (run 4). Comme toutes les images de la tâche MFLE sont des scènes de crues, nous supposons que la classe positive se distingue davantage par des caractéristiques à l'échelle d'objets (i.e. les classes d'ImageNet sont caractérisées par la présence d'objets), plutôt que par une reconnaissance holistique de la scène. La figure 2 montre des cartes d'activation (*Class Activation Maps* (Zhou et al., 2016)) d'exemples avec une forte activation positive ou négative selon des modèles InceptionV3 entrainés puis ajustés sur les tâches MMSat. Nous voyons que la classe positive est souvent associées à la détection de motifs d'eaux de surface. Ceci explique peut-être la performance modeste sur la tâche MFLE, où ces motifs sont alors peu discriminants. D'autres part, dans les deux tâches, les faux négatifs sont souvent associés à la présence d'éléments trompeurs dans l'image (e.g. micro pour NITD, tas d'ordures pour MFLE dans la figure 2).

5 Conclusion

Nous avons testé plusieurs approches à la détection de la sévérité des crues dans les articles d'information. Nous avons souligné la pertinence du transfert de modèles pré-entrainés sur des tâches proches qui disposent de grands volumes de données d'entrainement. De plus, nous avons observé que globalement, le gain est d'autant plus grand que ces données d'entrainement sont proches de la tâche d'intérêt. Nous avons également pu attribuer un bénéfice à un modèle

à données mixtes, plutôt qu'à traiter chaque modalité séparément. Ce dernier gain demeure cependant limité en comparaison de celui entrainé par le transfert de modèles.

In fine, si la performance obtenue sur la tâche NITD est satisfaisante (score F1 proche de 90%), il semble qu'une marge importante de progression demeure pour la tâche MFLE. Dans un esprit pragmatique, nous avons opté pour des modèles de classification binaire usuels, adaptés au déséquilibre des tâches de l'atelier grâce à la pondération d'instance et l'utilisation du score F1 en tant que métrique de validation. Vu le déséquilibre de MFLE ($\approx 3\%$ d'instances positives), une piste possible serait l'emploi de modèles proches de la détection d'anomalies (Perera et Patel, 2018).

Références

Audebert, N., C. Herold, K. Slimani, et C. Vidal (2019). Multimodal deep networks for text and image-based document classification. *arXiv :1907.06370 [cs]*.

Bischke, B., P. Helber, S. Brugman, E. Basar, Z. Zhao, M. Larson, et K. Pogorelov (2019). The Multimedia Satellite Task at MediaEval 2019. In *Proc. of the MediaEval 2019 Workshop*.

Bruneau, P. et T. Tamisier (2019). Transfer learning and mixed input deep neural networks for estimating flood severity in news content. In *Proc. of the MediaEval 2019 Workshop*.

Bulat, A. et G. Tzimiropoulos (2016). Human pose estimation via Convolutional Part Heatmap Regression. In *European Conference on Computer Vision*, pp. 717–732.

Chen, P., Y. Sharma, H. Zhang, J. Yi, et C. Hsieh (2018). EAD : Elastic-Net Attacks to Deep Neural Networks via Adversarial Examples. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Deng, J., W. Dong, R. Socher, L. Li, L. Kai, et F. Li (2009). ImageNet : A large-scale hierarchical image database. In *IEEE CVPR*, pp. 248–255.

Graves, A. (2012). Supervised Sequence Labelling. In *Supervised Sequence Labelling with Recurrent Neural Networks*, pp. 5–13. Springer Berlin Heidelberg.

Hadsell, R., S. Chopra, et Y. LeCun (2006). Dimensionality Reduction by Learning an Invariant Mapping. In *IEEE CVPR*, Volume 2, pp. 1735–1742.

Kingma, D. et J. Ba (2015). Adam : A Method for Stochastic Optimization. In *International Conference for Learning Representations*.

Kiros, R., R. Salakhutdinov, et R. Zemel (2014). Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models. In *NIPS Deep Learning Workshop*.

Limsopatham, N. et N. Collier (2016). Bidirectional LSTM for Named Entity Recognition in Twitter Messages. In *Proceedings of the 2nd Workshop on Noisy User-generated Text*, pp. 145–152.

Lopez-Fuentes, L., J. van de Weijer, M. Bolaños, et H. Skinnemoen (2017). Multi-modal Deep Learning Approach for Flood Detection. In *Proc. of the MediaEval 2017 Workshop*.

Pennington, J., R. Socher, et C. Manning (2014). Glove : Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543.

Perera, P. et V. Patel (2018). Learning Deep Features for One-Class Classification. *arXiv :1801.05365 [cs]*.

Poplin, R., A. Varadarajan, K. Blumer, Y. Liu, M. McConnell, G. Corrado, L. Peng, et D. Webster (2018). Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning. *Nature Biomedical Engineering 2*(3), 158–164.

Sandler, M., A. Howard, M. Zhu, A. Zhmoginov, et L. Chen (2018). MobileNetV2 : Inverted Residuals and Linear Bottlenecks. In *IEEE CVPR*, pp. 4510–4520.

Simonyan, K. et A. Zisserman (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference for Learning Representations*.

Szegedy, C., V. Vanhoucke, S. Ioffe, J. Shlens, et Z. Wojna (2016). Rethinking the Inception Architecture for Computer Vision. In *IEEE CVPR*, pp. 2818–2826.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, et I. Polosukhin (2017). Attention is All you Need. In *Advances in Neural Information Processing Systems 30*, pp. 5998–6008.

Yang, Z., D. Yang, C. Dyer, X. He, A. Smola, et E. Hovy (2016). Hierarchical Attention Networks for Document Classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pp. 1480–1489.

Zhang, X., J. Zhao, et Y. LeCun (2015). Character-level Convolutional Networks for Text Classification. *arXiv :1509.01626 [cs]*.

Zhou, B., A. Khosla, A. Lapedriza, A. Oliva, et A. Torralba (2016). Learning Deep Features for Discriminative Localization. In *IEEE CVPR*, pp. 2921–2929.

Zhou, B., A. Lapedriza, A. Khosla, A. Oliva, et A. Torralba (2018). Places : A 10 Million Image Database for Scene Recognition. *IEEE PAMI 40*(6), 1452–1464.

Zhou, P., W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, et B. Xu (2016). Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, pp. 207–212.

Summary

This paper describes the application of deep learning approaches that use textual and visual features to flood severity detection in news content. In the context of the MediaEval 2019 Multimedia Satellite (MMSat) workshop, we test the value of transferring models pre-trained on large related corpora, as well as the improvement brought by dual branch models that combine embeddings produced by mixed textual and visual inputs. We compare these model variants using data distributed in the context of the MediaEval MultiMedia Satellite (MMSat) 2019 workshop. The results disclosed in this paper were presented at the workshop: the present paper significantly extends the concise technical notes bound to the results obtained on the test sets provided by the workshop organizers.