

DIFFERENTIABLE NEURAL COMPUTER

Studio e sperimentazione nella Cross-Domain Sentiment Classification

Corso di Laurea Triennale in Ingegneria e Scienze Informatiche

Sessione di Laurea II

Anno Accademico 2016-2017

Relatore: Chiar.mo Prof. Gianluca Moro

Correlatori:

Dott. Ing. Andrea Pagliarani,

Dott. Ing. Roberto Pasolini

Presentata da :

Diego Pergolini

INTRODUZIONE ED OBIETTIVI DELLA TESI

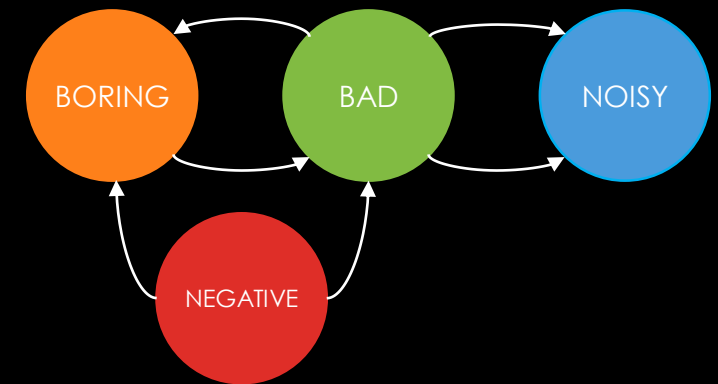
- La mole di dati testuali prodotta dai Social Network ha fatto crescere l'interesse verso la **Sentiment Classification**.
- In ambito commerciale sono preminenti gli scenari **Cross-Domain**.
 - Applicazione in domini dove non sono disponibili dati etichettati.
- Pochi sono, però, i lavori che fondono la **Cross-Domain Sentiment Classification** e le **reti neurali**.
- Obiettivo: applicare **Differentiable Neural Computer**, per la prima volta, alla Sentiment Classification.
 - Confronti con lo stato dell'arte tramite dataset reali di benchmark: **Amazon Product Data** e **Stanford Sentiment Treebank**
 - Applicazione di tecniche di **transfer learning** esplicite, come il **fine-tuning**, valutandone efficacia e praticità

SENTIMENT CLASSIFICATION

- Definizione di **Sentiment Classification**:
 - *Attribuire un valore di polarità ad una porzione di testo più o meno vasta, con campi di applicazione che possono variare da interi documenti a singoli tweet.*
- Esistono più tipologie di Sentiment Classification:
 - La **Binary** Sentiment Classification in cui il testo viene valutato cercando di capire se possiede una connotazione negativa o positiva.
 - La **Multi-Class** Sentiment Classification che permette di scegliere tra più classi (es: 5 nel caso del rating di prodotti Amazon), consentendo un ranking più espressivo
- Negli scenari **In-Domain** un modello viene addestrato utilizzando documenti pre-categorizzati di un certo dominio e testato su altri documenti appartenenti allo stesso dominio.
- Nei casi **Cross-Domain**, vengono addestrati modelli su un dominio (source), etichettato, ed utilizzati su un diverso dominio (target), non etichettato.
 - I domini considerati possono essere molto diversi tra loro, nasce quindi l'esigenza del **Transfer Learning**.

TRANSFER LEARNING

- Le tecniche di **Transfer Learning** si pongono l'obiettivo di colmare la distanza semantica esistente fra due diversi domini.
 - Si desidera applicare efficacemente il modello di conoscenza astratto dal dominio sorgente **S** su un dominio target **T** \neq **S**.
 - Fondamentale in ambito Cross-Domain, nei casi dove sono disponibili dati categorizzati solo del dominio **S** ma è richiesta l'applicazione nel dominio **T**.
- Vista l'**eterogeneità** del linguaggio naturale, è fondamentale riuscire a classificare correttamente testi contenenti termini mai presentati prima alla rete, ad esempio:
 - Viene addestrato un modello su **S**, dove i termini '**boring**' e '**bad**' sono spesso correlati ed assumono connotazione negativa.
 - Il modello viene quindi applicato al dominio **T**, dove compare il termine '**noisy**', mai classificato prima.
 - Come gestire questa situazione?



LE RETI NEURALI

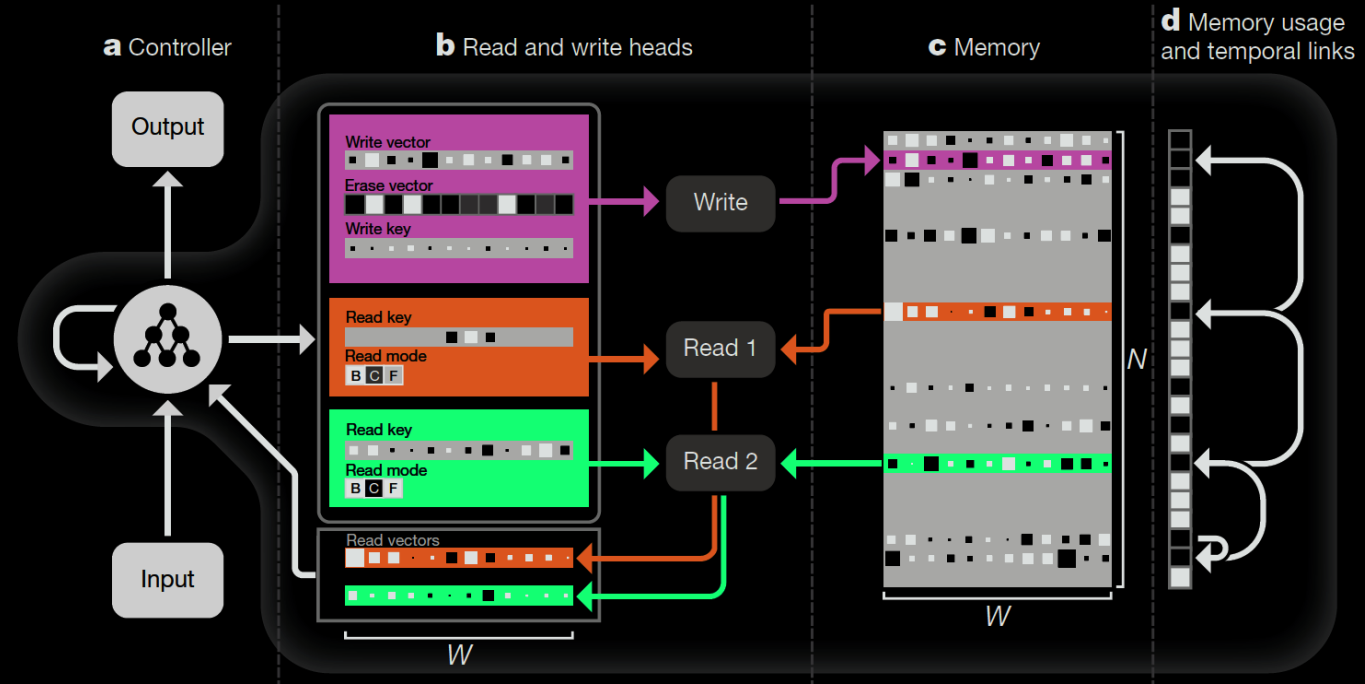
- La complessità di molte applicazioni reali rende quasi impossibile modellarle efficacemente tramite programmazione esplicita.
 - Tipicamente quelle applicazioni dove è richiesta una capacità di generalizzare ed astrarre concetti.
- Le **reti neurali** si propongono di superare tale limitazione:
 - Ispirate dai principi di funzionamento del cervello umano, sono composte da **neuroni** collegati l'uno all'altro ed organizzati in layers.
 - Ogni nodo della rete produce un **output** descritto da una certa **funzione di attivazione** applicata ai valori pesati ricevuti dai nodi di **input**.
- Tipicamente le reti neurali vengono addestrate per esempi:
 - I parametri della rete vengono regolati, attraverso **discesa del gradiente**, durante le varie **epoche** di addestramento al fine di minimizzare la distanza fra output prodotto e quello atteso in ogni **batch** elaborato.
- Il **Deep Learning** rappresenta un moderno approccio al ML:
 - Le reti sono costituite da molti livelli nascosti, tra i quali però, vengono applicate funzioni altamente non lineari al fine di catturare dipendenze complesse nei dati

DIFFERENTIABLE NEURAL COMPUTER

- Le **DNC** sono una nuova tipologia di rete neurale, evoluzione delle NTM, presentate nell'ottobre 2016 in un articolo su Nature.
 - Di forte ispirazione biologica, imitano per certi versi, il funzionamento dell'ippocampo
 - Apprendono per esempi(e sono **differentiabili**) come le reti **neurali**, ma capaci di memorizzare strutture dati complesse come i **computer**.
- Sono essenzialmente **RNN** dotate di una memoria auto associativa:
 - La memoria esterna viene gestita in totale autonomia, permettendo la memorizzazione di strutture complesse al fine di eseguire al meglio vari task.
 - Per analogia con le macchine di Turing, l'interazione con la memoria avviene attraverso delle '**testine**', il cui funzionamento dipende dal **Controller**.
 - I pattern individuati nell'addestramento sono indipendenti dalla dimensione della memoria, che è quindi espandibile, rendendo le DNC Turing-complete.
- Molteplici applicazioni e capacità di apprendere autonomamente algoritmi:
 - Individuare il percorso più breve tra due fermate di una metropolitana, navigare un albero genealogico cogliendo le relazioni di parentela o nel Question Answering.

ARCHITETTURA DELLE DNC

- **Controller:**
 - Si tratta di una rete neurale multi strato, accoglie gli input, gestisce la memoria esterna e produce gli output.
- **Read Heads:**
 - leggono contemporaneamente una parte della memoria. Possono utilizzare diverse modalità di lettura.
- **Write Heads:**
 - Definiscono un erase vector ed un write vector allo scopo di scegliere cosa eliminare e cosa scrivere in memoria.



- **Links e Usage vectors:**
 - Meccanismi avanzati che permettono di mantenere traccia dell'ordine di scrittura e dello stato di utilizzo delle locazioni di memoria.

INTERAZIONE CON LA MEMORIA

- Le DNC forniscono diversi meccanismi per interagire con la memoria:
- **Content attention:**
 - Il controller forma un vettore-chiave che viene comparato al contenuto di ogni locazione di memoria, basandosi sulla similarità tra i dati.
 - Meccanismo fondamentale per navigare strutture dati associative e dar luogo ad associative recall.
- **Temporal order attention:**
 - Una apposita matrice tiene traccia dell'ordine di scrittura (ed aggiornamento) dei dati in memoria, permettendo di recuperare sequenze nell'ordine corretto.
- **Memory allocation attention:**
 - Meccanismo utilizzato nelle procedure di scrittura in memoria, permette di andare ad utilizzare solo le locazioni di memoria libere, anche se non contigue tra loro.
 - Ad ogni locazione è associato un valore numerico che aumenta per ogni sua scrittura e diminuisce ad ogni lettura, permettendo di riallocare celle non più utili.

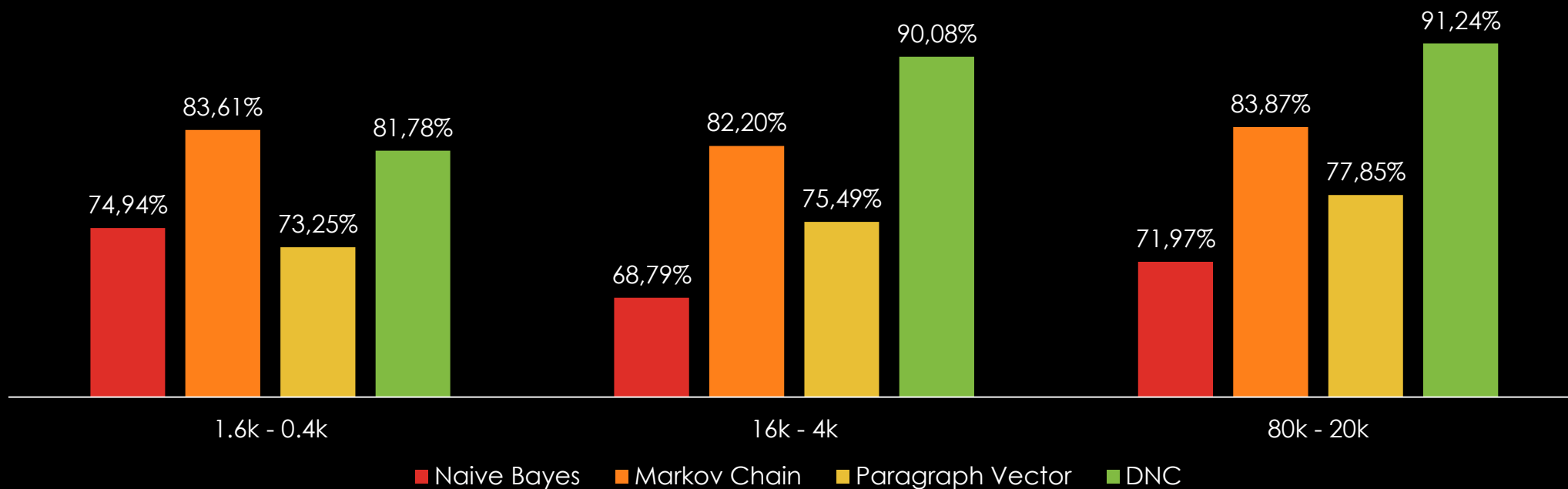
ESPERIMENTI

- Per valutare l'efficacia delle **DNC** nella Sentiment Classification sono stati pianificati diversi esperimenti, sia **In-Domain** che **Cross-Domain**.
- Esperimenti **In-Domain** su prodotti Amazon di 4 categorie:
 - Books (B), Movies and Tv show (M), Electronics (E), Clothing, Shoes and Jewelry (J).
- Esperimenti **Cross-Domain** tra tutte le possibile coppie di categorie
 - Applicazione del Fine-Tuning sia con 1000 che con 5000 recensioni a modelli pre-addestrati (con 80000 istanze) su un dominio diverso da quello target.
- Per la classificazione **a 2 classi** sono state usate 3 configurazioni di dataset di training e test, tutte con egual distribuzione tra le polarità:
 - 1.6k Training – 0.4k Test • 16k Training – 4k Test • 80k Training – 20k Test
- Classificazione a **2** e **5** classi su dataset comprendente più categorie:
 - **3600k** Training – **400k** Test • **3000k** Training – **650k** Test
- Sentiment Classification a 2 e 5 classi su **Stanford Sentiment Treebank**

PROGETTO ED IMPLEMENTAZIONE

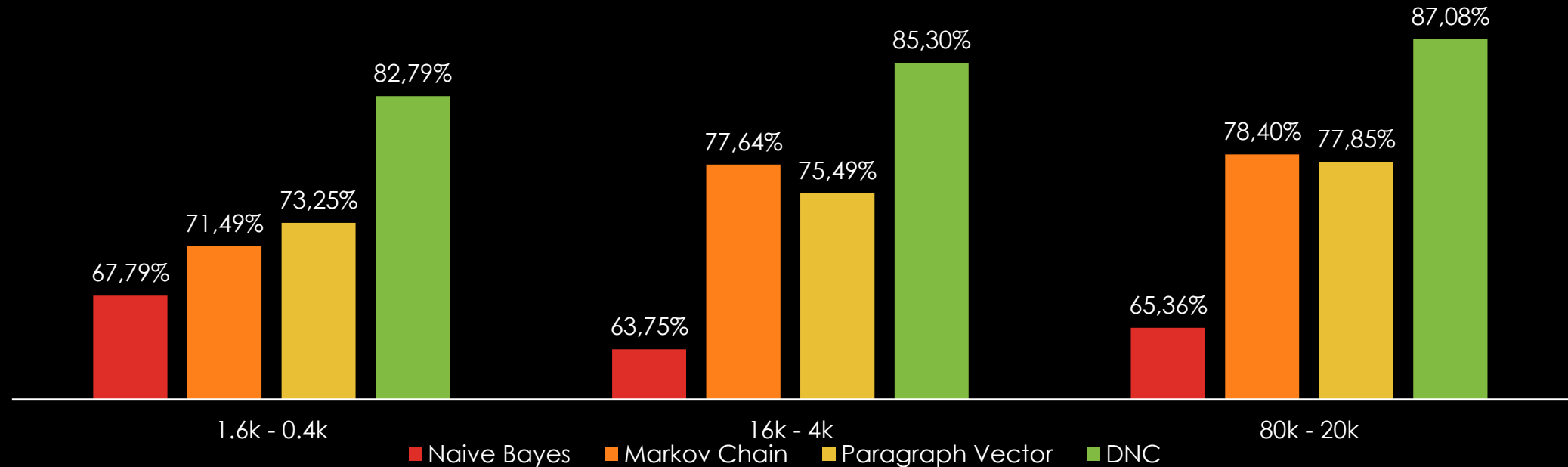
- Al fine di eseguire gli esperimenti preposti è stata utilizzata l'implementazione delle **DNC** sviluppata in **TensorFlow** da DeepMind.
- Le recensioni prese in esame, devono essere codificate in modo tale che la rete neurale possa fruirne:
 - Abbiamo deciso di utilizzare i **Word Embeddings**, in particolare un modello **W2V** addestrato sulle **Google News** del 2012/13 (Circa 100 miliardi di parole)
 - Questa codifica semantica, grazie alla distribuzione delle parole nello spazio di rappresentazione, fornisce un'efficace base in scenari cross-domain
- Sono state sviluppate delle funzioni per la preparazione degli input, per :
 - **Scelta** delle recensioni per training e test in modo da bilanciare la distribuzione tra classi.
 - **Codifica** delle recensioni, preparazione delle **label** e relative **maschere**
 - Divisione in **batch**, fornendo un iteratore, al fine di minimizzare l'utilizzo di memoria.
- Sviluppo del codice relativo all'addestramento e test della rete neurale.

RISULTATI IN-DOMAIN



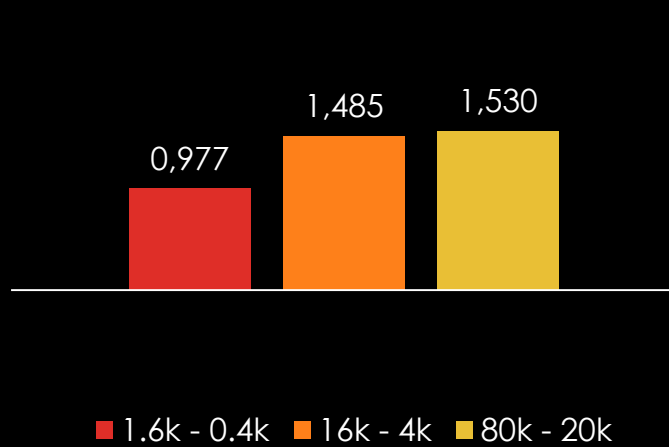
- Con 16k e 80k recensioni di training le DNC ottengono i risultati migliori.
- Tipica caratteristica degli approcci Deep Learning è la grande quantità di dati richiesta per costruire modelli adeguati
- L'incremento maggiore lo si trova passando da 1.6k a 16k istanze.

RISULTATI CROSS-DOMAIN



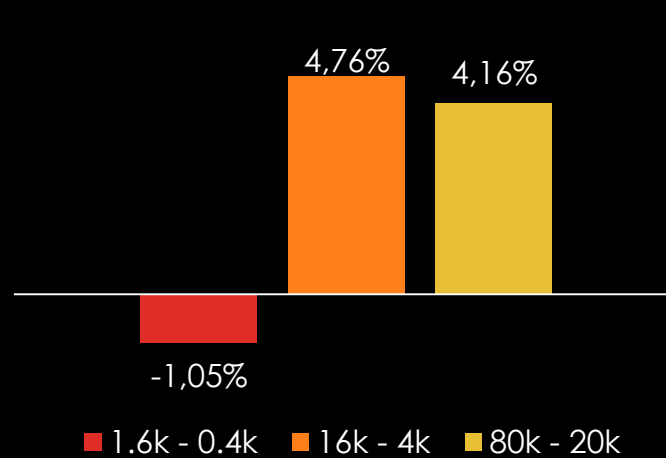
- Le DNC raggiungono risultati molto alti anche senza una fase di transfer learning esplicita.
- Anche con pochi dati di training il modello risultante è piuttosto generale
- Incremento costante dell'accuratezza all'aumentare dei dati utilizzati

TRANSFER RATIO, TRANSFER LOSS E FINE TUNING



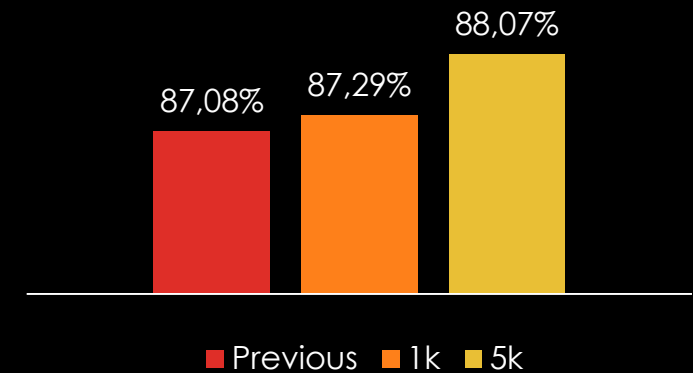
Transfer Ratio

- Questa metrica ci permette di notare come aumentando il numero di recensioni i modelli risultanti sono, in proporzione, più specifici al dominio sorgente.



Transfer Loss

- La percentuale di accuratezza persa in media durante il passaggio da un dominio all'altro è comunque contenuta.

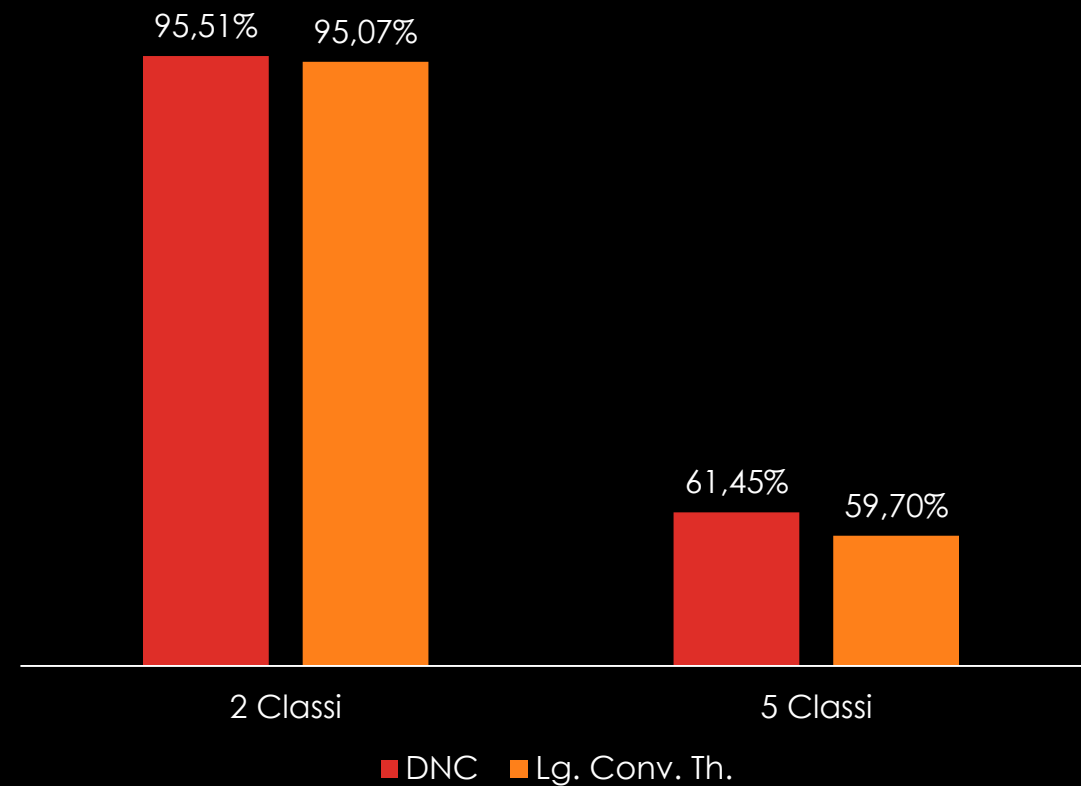


Fine-Tuning

- Applicando questa tecnica si ottiene un incremento, seppur contenuto, delle accuratezze. Migliorando i, già ottimi, risultati.

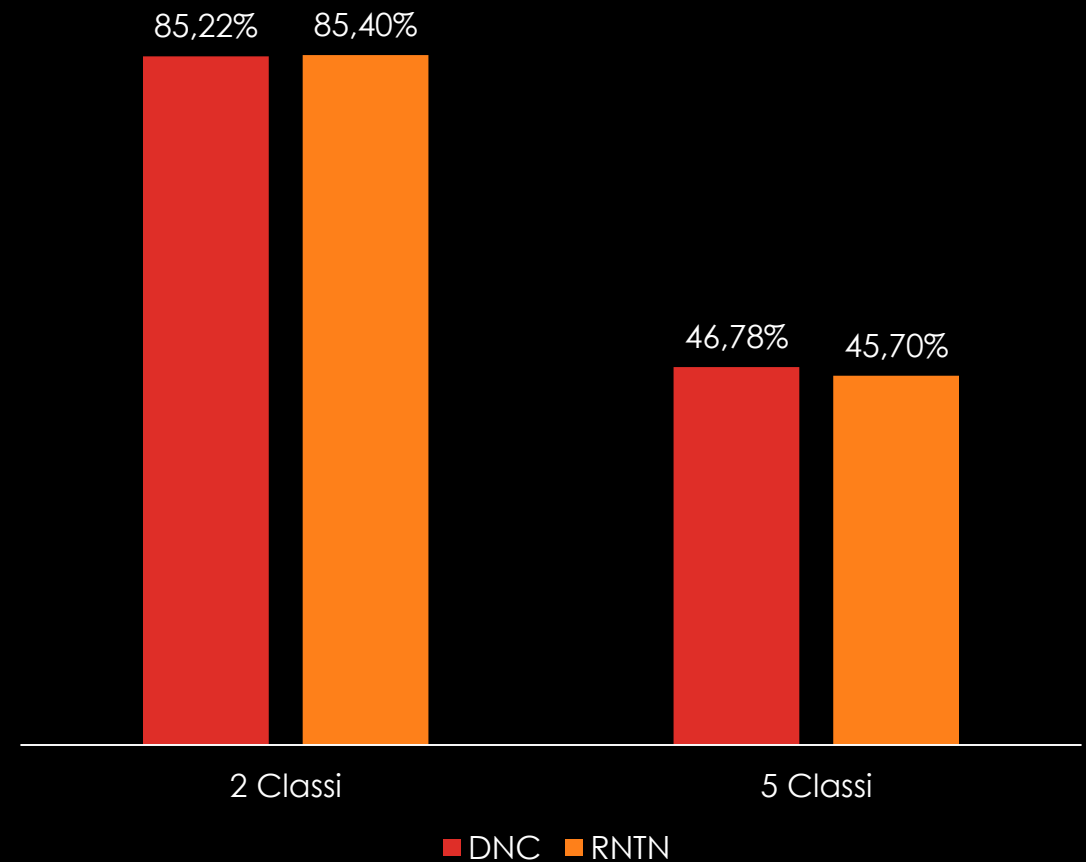
RISULTATI LARGE-DATASET

- Confronto dei risultati ottenuti dalle DNC con quelli rappresentanti lo stato dell'arte di Zhang et Al. (utilizzando CNN)
- In entrambi gli esperimenti le DNC fanno segnare risultati migliori.
- Risultati sorprendenti se si considera che le recensioni provenivano da 4 diversi domini.



RISULTATI STANFORD SENTIMENT TREEBANK

- Confronto dei risultati ottenuti dalle **DNC** con quelli prodotti da Socher et al., 2013, utilizzando una **Recursive Neural Tensor Network**.
- Nella classificazione a 2 classi risulta migliore la RNTN mentre in quella a 5 classi le **DNC**.
- Le **RNTN** sono però un metodo specifico per single sentences, mentre le **DNC** hanno uno spettro d'azione più ampio.
- Il limitato numero di istanze di training (rispettivamente **6920** e **8500**) può influire sulla accuratezza del modello.



CONCLUSIONI

- L'applicazione delle DNC alla Sentiment Classification ha dimostrato non solo la generalità di tale tecnica ma anche le sua potenzialità nell'individuare correttamente le polarità di documenti non strutturati.
- I risultati ottenuti nei dataset Amazon sono ottimi:
 - Accuratezze molto alte nei test In-Domain, migliori degli altri metodi considerati.
 - Grande capacità di applicare la conoscenza appresa da un dominio sorgente su un diverso dominio target, anche a fronte di domini molto diversi tra loro. Rivelandosi quindi un ottimo approccio per gli scenari Cross-Domain.
 - Risultati superiori allo stato dell'arte negli esperimenti con milioni di istanze provenienti da domini eterogenei.
- Sono stati ottenuti risultati simili a quelli in letteratura su un dataset molto utilizzato come lo Stanford Sentiment Treebank
 - Vista la profonda diversità del SST dai dataset Amazon, ed i comunque buoni risultati ottenuti, le DNC si dimostrano efficaci nel classificare sia singole frasi che interi documenti.
- Le DNC sono perciò in grado di svolgere egregiamente task complessi come la Sentiment Classification.