Salve a tutti, sono davide montagno, presento l’assignment 2 ovvero l’identificazione di più topic in un’immagine tramite l’uso di LDA.

Il primo passo è stato quello di estrarre tutti i descriptor; la prima osservazione che posso fare è che MSER è molto più preciso di SIFT ed il numero di keypoint estratti è minore. Dopo aver concatenato tutti i descriptor e tramite il KMENS, è stato costruito il dizionario globale. Da questo tramite il vector quantization è stato possibile ottenere tutte le word per ogni insieme di descriptor che compongono l’immagine, realizzando un istogramma generale formato dalla word ottenuta dal vector quantization e la sua relativa frequenza; tale operazione è la realizzazione della Bag of Visual Word per ciascuna immagine, utilizzata per il training dell’LDA. Infine su ciascuna immagine presa dal test set è stata fatta una predizione utilizzando il modello precedentemente allenato in modo da estrarre tre oggetti: il primo estrae la distribuzione di probabilità che un immagine appartenga ai diversi topic; il secondo è un oggetto chiamato word\_topics mentre il terzo è chiamato phi\_Values. Quest’ultimi sono strettamente correlati tra di loro, difatti il primo restituisce una lista dove per ciascuna word è formata una lista di probabilità per ciascun topic (quindi è possibile avere un duplicato del topic con diversa probabilità ma essendo messe in ordine decrescente prendiamo sempre il primo valore), mentre la seconda fornisce la probabilità massima che ciascun topic ha per ciascuna word. Nello snippet è stata utilizzata la prima per semplicità, ma è stata verificata la funzionalità pure con i phi\_values.

Una volta verificato che il modello rispondeva a quello che volessi realizzare sono passato alla fase di model selection. Dapprima ho utilizzato i parametri di alpha forniti da gensim (auto e symmetric). in quest’ultimo viene usata una prior fissata e normalizzata, mentre nel primo viene appresa in modo asimmetrico dal corpo che si sta utilizzando (l’intero set di training dunque). Non soddisfatto dei risultati, ho eseguito il tuning di alpha e del numero di topic. Inizialmente ho provato con alpha alti compresi tra 10 e 1, ma il modello essendo poco specializzato generalizzava male poiché con un alfa alto i documenti sono formati da più topic e quindi la distribuzione sui vari topic risultava quasi uniforme, ho deciso pertanto di abbassarlo con valori compresi tra 0.01 e 0.09.

Il modello migliore ottenuto, precisando che i risultati sul training e tra i diversi test erano molto instabili, ha un numero di topic pari a 12 e alpha a 0.05. Come possiamo vedere i keypoint dell’erba assumono colore blu, mentre la mucca è identificata con l’arancione e alcuni keypoint vengono identificati di blu come l’erba. La faccia assume due colorazioni azzurrino e giallo (quest’ultimo probabilimente identifica i contorni della faccia). Infine l’auto e la bici vengono identificati quasi con la stessa colorazione, difatti alcuni punti della bici sono classificati con un verde scuro.

Per finire, ho cercato di capire il perché dell’instabilità del modello, questa probabilmente dovuta sia al fatto che il dataset fosse molto piccolo e sia alla forte ricorrenza di pattern simili all’interno delle immagini. Inoltre, a seguito di alcuni esperimenti ho notato che probabilmente il numero di cluster fosse troppo elevato rispetto ai keypoint estratti dal dataset di training (ho provato con 200 e 300 cluster e sembrava funzionare meglio). Ho cercato di valutare il modello tramite la perplexity ma dopo svariate ricerche online ho appreso che il modo più utilizzato è la *coherence (ovvero il grado di semantic similarity tra le word con alto score nel topic) ottenendo quindi* i topic con il più alto score di *coherence.* Non essendo abile a interpretare questi dati non ho avuto modo di verificare che il modello funzionasse correttamente.

Come ultimo commento ho iniziato con l’utilizzo della librearia scikitlearn. Quest’ultima però forniva soltanto la distribuzione di probabilità relativa ai topic sull’intera immagine, dunque ho cercato di aggirare il problema creando un istogramma per ogni sift per utilizzarlo come dataset di train per il modello LDA relativo alla librearia. La troppa sensibilità sui dati non mi forniva però risultati soddisfacenti dunque ho preferito usare la librearia gensim.